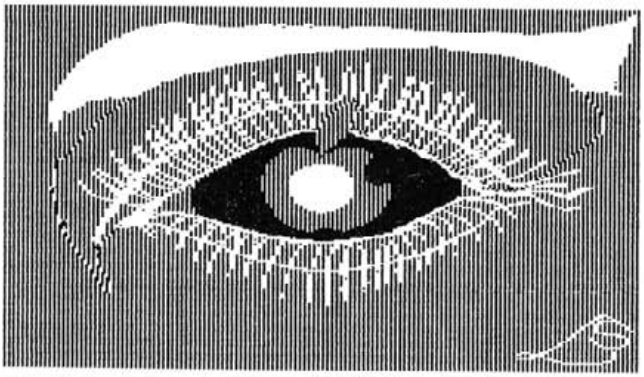
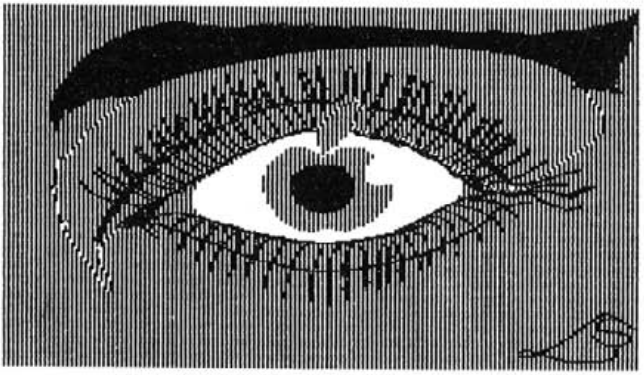
hard core

THE JOURNAL OF THE BRITISH APPLE SYSTEMS USER GROUP

AUGUST 1982 £1 VOLUME 2 NO 4





For everything **apple .. ring**

LUH Computer Services Ltd.

108 THE PARADE HIGH STREET WATFORD WD1 2AW
Telephone: Watford (0923) 29513

* Business Software * Professional Courses * Pascal *
* Recreational Software * Interfacing * Cromemco Systems *
* Level 1 Service Centre * Utilities * Colour Monitors *
* Supplies * Accessories * Enthusiasm * Open Tues - Sat *

MICROSOURCE SUMMER SALE

We have a few copies of books which are not in pristine condition. They have been damp at some stage and the pages have ended crinkley. They are not stuck together and are quite legible.

TWO COPIES ONLY of COMPUTER GRAPHICS PRIMER

normal price ~~£10.45~~ sale price £5

FIVE COPIES ONLY of BENEATH APPLE DOS

normal price ~~£11.95~~ sale price £6

WILDCARD

as advertised in the insert with this issue, is available

for ~~£99~~ incl VAT P&P

FROM
MICROSOURCE

1 BRANCH RD, PARK ST
ST ALBANS, HERTS

MICROSOURCE

ONE ONLY

OMNIBOARD

The "everything" I/O board from EDLECTIC SYSTEMS
Parallel I/O with handshaking
24 hr real time clock with alarm
2K eprom with graphics I/O and screen editor software
6 2K Eprom sockets
RS232 I/O (software driven but very basic and without handshaking)
256 byte PROM supervisory software.
Generate full ASCII character set from keyboard. Character rotation etc.

Extensive documentation and demonstrations disk[illegible] including SOFT CHARACTER EDITOR.
Much much more and developable to your own ends.

ONLY ..... ~~£95~~ incl VAT

ALL PRICES INCLUDE VAT UNLESS SPECIFIC MENTION IS MADE. ALL PRICES INCLUDE P&P

MICROSOURCE
Tel: Park Street (0727) 72917

1 Branch Road,
Park Street
St. Albans.

hardcore

THE JOURNAL OF
THE BRITISH APPLE SYSTEMS USER GROUP
P.O. BOX 174 WATFORD WD2 6NF

EDITED BY DAVID BOLTON
AND TONY WILLIAMS

Editorial assistants: Tony Game, John Sharp, Bob Raikes and many others.

BRITISH APPLE SYSTEMS USER
GROUP COMMITTEE

NORAH ARNOLD	Chairman
JOHN SHARP	Secretary
FRAN TEO	Treasurer
JIM PANKS	Membership Secretary
BOB RAIKES	Events Organiser
JOHN ROGERS	Software Librarian
JOHN RODGER	Librarian /Information Officer

JOHN WELLSMAN, JOHN MARTIN, TONY WILLIAMS, DAVID BOLTON, MIKE PRESTON

BANKERS

National Westminster Bank Ltd.,
Brompton Square, London SW3 1HH.

MEMBER OF THE
INTERNATIONAL APPLE CORE

COPYRIGHT (C) - The contents of this journal are copyright of the British Apple Systems User Group and / or the respective authors. However permission will normally be granted for non-commercial reproduction by user groups affiliated to the International Apple Core, provided the author and source are properly credited.

The opinions and views expressed are those of the various contributors, and are not necessarily supported by the British Apple Systems User Group.

CONTENTS

4	Editorial Chairman's Corner	Norah Arnold
5	Readers' Letters	
10	Courses, Meetings	R.T.Raikes
11	More letters Pascal Notice	
12	UCSD Pascal Course	Leo Crossfield
16	Machine Code Course	R.T.Raikes
17	Education Column	Norah Arnold
18	BASUG Toolkit No 2 Get Hex	Cliff Wootton
20	Annual General Meeting	
23	Amateur Radio	G8LDR
24	Nicelister Headings	M.J.Parrott
25	French Apple Keyboard	Tony Williams
26	Herts Group	
27	Applewriter II	J.Panks, Richard Teed
29	More letters	
30	Leicester Laughs	
31	Time Machine Snafu Time Again	Ian Pawson Simon Brown
32	Beginners' Pages	John Sharp
33	Eamon Problems Solved	John Martin
34	Input Fix	Simon Brown
37	Letters	
39	Add Auto Repeat	
41	Expanded HIRES	Richard Teed
43	Standard Deviance	R.D.Purves
44	Now you see it	Neil Lomas
45	Type Right Review	Martin Rogers

Cover Illustration: The Apple of your Eye: Stolen from A.U.G.E., the German Apple User Group, because we liked it.

INTERNATIONAL APPLE CORE TM

MEMBER OF THE
INTERNATIONAL APPLE CORE

EDITORIAL

Two things are immediately striking about this issue of Hardcore. The first is the new format. In an attempt to catch some of the anticipated flak, let me blame it all on member Jim Watson. He it was who ran up a reduced mockup on his photocopier to show that it could be done, producing considerable savings without diminishing the quality. We were convinced. Since a sizeable proportion of your membership dues go toward producing the magazine, the committee could not brush aside his suggestion simply because we might prefer the large format. Economy rules, I'm afraid. Discussion raged into the committee nights, and it was argued that the A5 format in any case fits in with the norm for user clubs, that reduction by pruning our costs enables us to be less dependent on advertising and to run a tighter ship. The decision was taken with some regret, since we know that many members like to keep their bookshelves straight and abhor abrupt changes. Our apologies to you all.

The other striking feature is the preponderance of your letters we print this time. On this we are quite unrepentant. Hardcore exists in large measure as a forum for your views, and we believe that publishing letters is the best way for that. Your letters tell us most about how you use your computers and the problems you encounter. To clinch matters read the contribution from Mr. Santos on power supply problems. It could save your Apple and sav e you hundreds of pounds.

Thanks also to the members throughout the world who have sent in articles. Two things bother us. One, what to do with mammoth listings extending to fifteen columns or more, sometimes consisting largely of esthetically pleasing but wasteful white space. They are too big to publish and yet we don't want to bury them in the archives. What to do? We are still thinking. Two: Would thoughtful contributors who send in hardcopy printouts in Hardcore format please never, never paste them up. Leave it to us. Just send unsliced galleys. And please, accompany them by the WP file on disk since more often your (and our) spelling is not as good as we would wish. But, keep it coming.

Tony Williams

CHAIRMAN'S CORNER

by Norah Arnold

It comes as something of a shock to me to find myself writing the Chairman's Corner. It doesn't seem very long ago that I waited until my husband was safely out of the house before making my first tentative approach to the Apple II. Two hours later, having become thoroughly addicted in the meantime, I can recall wondering what effect this machine might have on the life of myself and my family.

BASUG has certainly come a long way since I attended my first meeting towards the end of 1980. On reflection, I have quite enjoyed putting all those Hard Cores into envelopes, sticking them down with sponges that always seemed too dry or too wet and glue that never appeared to be sticky enough until your fingers got stuck together.

This time last year the membership of BASUG was approaching 600, now it is almost double that figure. The task of looking after membership records has been taken over by a new Membership Secretary, Jim Panks. At the time of writing, I understand that Jim is in hospital so I would like to take this opportunity to wish him a speedy recovery.

BASUG has had a stand at many shows during the last year, and all those who have given up their time to help at the shows deserve a vote of thanks from the group as a whole. A special thank you should go to Frances Teo for all her hard work (in spite of the heat) at Apple '82 at Slough.

Bob Raikes is in the process of organizing the BASUG stand at the P.C.W. Show, 9th to 12th September at the Barbican, and he has some new ideas for getting BASUG around the country a little more than in the past. I understand that something is being planned for September 25th at Birmingham Polytechnic. I hope Bob will get a great deal of support from BASUG members in the midlands.

John Sharp continues to do battle with a never ending mountain of post, kept going by his good nature and eternal optimism. Keep up the good work, John!

Finally I would like to welcome the new members of the committee and also thank the retiring chairman, Frank Kay, for all the work he put in during BASUG's formative period.

reader's' letters'

Sydney,Crewe.

Dear David,

In the June issue, you asked for readers' views on the magazine presentation. I bought my 737 because I was impressed with its proportional character set and, on its own, I think it perfectly acceptable. Of course, the individual characters do not stand comparison with those of a good daisywheel, but good presentation involves more than pleasing letter shapes.

The Star Trek instructions were printed with a very pleasing typeface, but it was made more difficult to read because it was made more proportionately spaced and justification was done by quantum jumps in the gaps between words. The same criticism can be levelled at articles printed with the Epson, although these do not seem quite as bad, possibly because the narrow letters fill the space better.

The worst example I find was the Apple /// article in April. This was printed on a dot-matrix printer (which?) using a proportional set but without the appropriate spacing. The result was that the letters were so far apart that one virtually had to read the words letter by letter and assemble them.

If you can afford the expense of a daisywheel with proportional spacing and microspace justification (and the expensive word processor to utilise it), by all means make the change. If not, then stick to the 737, which anybody can use with a £3 word processor from the library.

A final advantage of the dot-matrix is that you don't have to resort to Letraset for one-off titles - but for a permanent title, isn't it time to move the apostrophe in "READER'S LETTERS" to the right side of the "S"?

Yours sincerely,

Neil Lomas.

(Ed. What can we say to that? Let's try this: we think of you all as individual s, and very singular individuals too, each of whom is likely to contribute more than one letter. No, it doesn't convince me either, we'll have to change it)

Waterford, Ireland

Friends,
I have the answers to two queries raised in Hardcore June 1982.

1- John Kleenan in 'Using DOS..",page 18, wonders about the use of location $76. This is the Applesoft Basic current line-number high byte, and under normal circumstances it is $FF when there is no Applesoft program running. Unless you use Poked line numbers greater than 65279 (which cannot be entered directly), when your program is running the, contents of $76 will not be $FF, and DOS tests this for all the commands which can give a NOT DIRECT COMMAND error - but only when the Basic is Applesoft. The corresponding location for DOS when Integer is active is $D9, which should be less than $80 for any legal line number while the program is running. So do:-

D9:0 to turn on textfiles.... and
D9:FF to turn off if Integer is active.

On the last line of page 18 the addresses should be 9D5A, 9D5C, 9D5E and not 905A etc.

2- Page 35, Graham Ruben's request C: patch to prevent OPEN from creating non-existent files: change the contents of $A923 from $23 to $22. Thus from Applesoft use POKE 43299,34 to do this, but don't forget to POKE 43299,35 if you want to OPEN a new file for WRITE.

Generally, for those working in this area, get 'Beneath Apple DOS'. There are a few errors in it, but it is an enormously valuable book.

Yours

Hugh Dobbs.

Poole, Dorset

Dear David

I feel that I must write in response to the letter from John Gribbin that appeared in the June issue of Hardcore. I did not buy my Apple solely to do a bit of wordprocessing, or solely to do a bit of anything else. I bought my Apple because I am a computerphile (sic), and the Apple was the best all round machine that I could afford. I am interested in as many aspects of the Apple as I can possibly squeeze in to my spare time. I welcome the diversity that BASUG currently supports, and welcome any attempts to extend the boundaries of the group's activities. I joined BASUG because I believed that it would enable me to contact people like myself, and knowledge and information would be spread around. That I have failed to have much contact with such people is my fault, not BASUG's. I would have thought that most of the members are hobbyists, and I would suggest that the single minded people such as Mr. Gribbin should form the separate group that he proposes (or stick to reading Windfall).

As regards to the courses, there will always be a problem of attendance when the members are spread all over the country, and when the cost of the activities has to come from their own pockets. I wanted to attend the Pascal course, but the dates were inconvenient. Please, please keep trying, it's worthwhile.

There is a facility that a group such as BASUG would clearly benefit from, and that is a 'Hotline', such as 'Call A.P.P.L.E.' has. I feel that this would be very beneficial, many is the time that I have had a problem that I was sure someone would know the answer to, but I had no-one to ask. I wonder how other members feel about such a facility ?

Having said my bit, I would like to reply to a point raised by Graham Rubens in the same issue. In his letter, Graham asks for a patch to stop non-present files being opened automatically. Rather than patch DOS for something that is not a bug, I would suggest an alternative approach. In the program, monitor all errors (using ONERR) and perform a DOS operation that will return the 'File Not Found' error (e.g. UNLOCK) before trying the open. If the File does not exist, proceed, if it does, open it.

Finally, may I take this opportunity to enquire whether anyone has an old (non-Autostart) monitor rom for sale.

Yours

Bob Phillips

P.S. It is not time to turn to daisywheel printers, some of us would still like to be able to purchase our first dot matrix printer.

Stevenage

Dear Mr. Bolton

I would like to take the opportunity of joining BASUG to say that your group appears to be playing its part in furthering the aims of Information Technology Year '82. Your bi-monthly magazines have provided some interesting answers to a few problems.

I would like to relate a problem that I have been experiencing when using the Apple Pascal Language System (Version 1.1) EDITOR on my ITT 2020 System. The CTRL-E enable reverse video mode and keyboard upper and lower case toggle, and the CTRL-W enable reverse video and set keyboard to upper case for next character only features DO NOT work as documented in the Operating System Manual Addendum. These features should provide pseudo upper and lower case - similar to Applewriter/SuperText II and other text editors - text input. These features however, do work correctly on an Apple II.
So far, I have found that this is the only software for the Apple to malfunction on an ITT 2020 when using upper to lower case change feature.
I hope that this problem does not frighten any more ITT 2020 owners into changing over to the APPLE, but I also hope that solutions can be found more speedily.

I have a lower case adapter and a modified version of PASCAL 1.1 which displays commands in Upper and lower case - a marked improvement on just Upper and Inverse Video to represent Upper case. However, this does not affect my inabilty to input lower case on my ITT 2020 - the problem exists on an unmodified machine too.
I hope that the above information will prove useful to the User Group's Membership.

Yours Faithfully,

Joe Gardner.

Dudley, West Midlands.

Dear Sir,

Having just joined the ranks of BASUG, and duly read the Feb. and Apr. issues of Hardcore, it occurs to me that a number of points were raised which merit attention.

Ladies first, and Miss Chuah may care to try the harp in getting past the giant (in Wizard & Princess), assuming she has managed to find out how to get it across the bridge (no mean feat in itself). A friend and I managed to 'crack' this particular adventure in a few evenings, so if she'd care to pass on her address via yourself, I'd be happy to elaborate on the above. I confess to being addicted to adventure games, and would dearly like to know how to make Dracula appear in Adventure International's 'The Count'. I've found the crypt - fixed his coffin - and lie in wait with the hammer and stake, but can't find out how to make him appear....any offers? Can one use 'Wizardry' without Pascal?

With regard to Mr Lightowler (G6BGN), my own call sign is G3ZPF, and I have found about 25 amateurs using the Apple. I'd be glad to know if he's found a cure for the RF interference radiated by the Apple. If he has the American SSTV program, and found (like myself) that it doesn't work properly in Europe, then I have a 'fix'. I also have a Hellschreiber program (Hellschreiber is the German WW2 version of our teleprinter).

As I've had my Apple for some time now, I've come across several routines and bugs which may be unknown to other members. To avoid repeating stale news I'll sumarise them briefly and go into details if asked.

1 - There is a bug in the later version of Renumber, even on the DOS 3.3 upgrade version. Again I have a fix.

2 - A friend tells me that the Laurence Hall disc copy prog. for his 8" discs works quite happily without mods on 5.25" discs under DOS 3.3 & 3.2.

3 - It is possible to alter the tail markers on DOS 3.3 sectors, so that although the disc boots, RWTS cannot be used to scan at the sector level. Following on from this, it is possible to get around the fact that pressing CNTR C during bootup stops turnkey systems in their tracks and hence add password entry to files. This may be of use in a office situation where several users have access to the micro, since although unauthorised users can use bit copiers to copy the disc 'in toto' they still can't use it without knowing the passwords. Anybody interested.....? Or is this all old hat?

Incidentally, in spite of the comments on FID under 3.3 (yes I'd got a 3.2 version running) not copying random access files, when I upgraded to DOS 3.3, I didn't have any problems that I've noticed. Perhaps the conditions to cause errors are quite specific, and if so then I'd like more details about them please.

Yours faithfully,

David J. Reynolds.

Campbeltown, Argyll.

Dear Sir,

The EPSON PRINTER PAGES in the June 1982 edition of Hard Core bring up again the use of the Applesoft TAB instruction. It has been pointed out in several journals recently that there are very few printers (none tht I have ever used) that can handle it other than as a synonym to SPC; so the EPSON range is not exceptional. The acceptable printer instruction is POKE 36,(column number), which will work on most printers. If this instruction is used it must be remembered that the first column is zero and not one. Line 690 of the CATALABEL program would read PRINT NAME$(Z);:POKE(36,80):PRINT NAME$(Z+INT(N/2)+P1 if the first character of the second string is actually required in column 80. POKE 36- may also be used for printing to the screen, but in this case the column number must not exceed 39 or else there is the probability that the start of the Applesoft program will be corrupted. It seems strange that APPLE do not emphasise this feature since the subject of Apple 11 tabbing is hardly ever out of the magazines.

There are two bugs in the original CATALABEL program that have been repeated in the June reprint. Lines 500 and 680 shoud be:-
500 FOR M=0 TO 29
(the program name is only 30 characters long)
680 FOR Z=1 TO N/2+P1
(let's print all the program names!)

Yours sincerely,

Hedley G. Wright.

Ramsey
Isle of Man

Dear Sirs,
Do you have any information on the Apple Dow Jones Portfolio System Order No A2D 007?

Yours sincerely

C W Sinclair
BASUG Member 8203518

Dear Mr Trackman,

I am using your Go-Between with the old Applewriter I coupled to my Centronics 737. This set-up has worked perfectly for me, so when Applewriter II came along I borrowed a copy and found many excellent features in the editor but have been frustrated to the Nth degree when trying to print out.

How do I get the 737 to change to proportional or condensed mode from within the program? I have done it by presetting prior to booting up Applewriter but this is not satisfactory. I have also found that several other features such as fill justification don't work.

I assume that this program wasn't really written for this printer. How does it work with the Epson?

The whole thing is at present a bit of a disappointment, but before I give up in despair and revert to AW I + G-B by Ian T, I wonder whether you or any other BASUG members can assist. It would be a super piece of software if it had a similar system of print formatting commands to the Go-Between - (Hint!).

Yours sincerely,

Peter Trinder.

(Ed: On Applewriter II in general please see Jim Panks/Richard Teed's article elsewhere in this issue. On your specific question, Applewriter II is protected proprietary software and Ian Trackman cannot simply go ahead and and make and market modifications to it simply because they might be a good idea. The whole area of copyright is a minefield, as you can imagine, and it is advisable to stay well clear. In any case, the requirement has now extended well beyond the Centronics 737 and now software is needed to drive a variety of proportional space printers, including the Qume. Such a driver would demand weeks of intricate programming and could never be obtained for 50p. (Apologies for the rather discouraging comment).

Leyburn,
N.Yorks

Dear David,

On re-reading my keyboard and display conversion in the June issue of Hardcore I noticed a minor fault in Fig.1 (my fault). The Schmitt trigger nand gates labelled IC2a - IC2d should be IC3a - 3d, but the wiring diagram is correct.

A more important point arose after I had submitted the article. I noticed that after being in use for some time, the changeover switch effect of pressing CTRL and Shift together did not always work. The reason for this is that the temperature rise within my Apple (which has four peripheral boards) causes the switching voltage for the Schmitt trigger inverter IC1d to vary. The cure is very simple. Just add a 1K Ohm resistor to the circuit board between pin 8 of IC1 and the ground line. This acts to pull down the voltage on this pin when a switch-over occurs, making the action more positive.

I would also like to pass on a little information to those who may not already have seen it. This concerns a modification to the Applewriter which appeared in the June issue of Creative Computing and allows underlining in conjunction with an Epson MX80 printer, and at the same time gives details of free space available within Applewriter (using a redundant, undocumented function). Well worth a read and no doubt available on loan from public libraries.

Finally I noticed some comments regarding Emphasised printing on the MX80 in recent issues of Hardcore and for what it may be worth my MX80 F/T prints emphasised characters (as my keyboard article) by sending the codes ESC E to the printer and this is cancelled by ESC F. The printer will double print if ESC G is sent and this is cancelled by ESC H. Both of these codes remain in force until cancelled by a reset or the CTRL code. It should be noted that some interface cards may not send control characters (such as ESC) without modification, and to get round this it will be necessary to POKE the code to the card I/O location (C0 n+80 for a CCS Centronics interface, n being the slot number).

Yours sincerely

Peter Blair

```
100  REM ************************
110  REM *  THIS PROGRAM MAKES  *
120  REM *       MORE SENSE     *
130  REM *    WHEN YOU RUN IT   *
140  REM *                      *
150  REM *     BY R.D. PURVES   *
160  REM ************************
170  DIM A(7100)
180 A = 0:B = 0:C = 0:D = 0:E = 0
190 F = 0:G = 0:H = 0:I = 0:J = 0
200  PRINT "HOW SLOW I AM TODAY!"
```

SPECIAL

The most versatile diskette filing system available.

This unique, modular system gives you safe, dust-free storage for your data while, at the same time, making it extremely easy to get to.

Sturdy, interlocking cases, each holding ten diskettes in a slide-out drawer, can be arranged in rows or stacks of almost any size to meet your space requirements.

As your diskette library grows, so does your storage/filing system. Any number of the modular cases can be interlocked to form a unified and convenient filing system. And, more cases can be added at any time — either horizontally or vertically.

And goes with you

When you travel, individual cases make perfect diskette carriers, and are easily removed without disturbing the rest of the system. Additionally, cases can be removed for storage in a fireproof safe, or easily rearranged to meet new space requirements.

Each case comes with two pressure sensitive identification labels. Label surfaces can be typed on or marked on with pen or pencil to show, at a glance, which diskettes the case contains. So, you can quickly find the diskette you need — when you need it

The ultimate in diskette storage, protection and retrieval

£2·75 INCL VAT

100 SUPERB LIBRARY BOXES AVAILABLE SUPPLIED ON REQUEST FREE WITH EACH BOX OF 10 AKKUTRAK DISKS SOLD (instead of normal library boxes)

OFFER

BRITISH APPLE SYSTEMS USER GROUP

P.O. Box 174, Watford WD2 6NF

COURSES, MEETINGS AND EVENTS

by R.T.Raikes

Those of you who have read the minutes of the recent A.G.M. will have noticed that I have taken over from Fran Teo as courses, meetings and events organiser.
People join BASUG for many reasons, but for me the most important aspect of the group has always been the chance to meet with other addicts and discuss problems, pick brains and see for myself some of the products which are available for my APPLE. I am lucky in living within 30 miles of Park Street village, where there is an active local group, and where many of the national meetings have been held.
My aim over the next year is to get BASUG out of the London/South East area for a substantial proportion of its meetings. In order to do this I will need a fair degree of help from members around the country.
There are four main changes to the regular monthly meetings that we currently hold.

1. Rather than have just a 2/3 hour talk on a particular topic, it is proposed that we hold Workshops which will last for a whole day e.g. from 10 to 6. At each venue we will have at least 2 rooms, one of which will be used for lectures, talks and demos. The other room will be a Systems room, where we will aim to have the maximum number of systems available for people to program, get help with problems, or just play the latest games.
2. These Workshops will generally take place on a Saturday, apologies to those who have to work then.
3. There will have to be a charge, approximately £3.50 is estimated, towards the cost of hire of halls etc.
4. In order for such Workshops to be a success, members will have to do their bit by bringing a system if at all possible (including a mains multiway socket if possible), and by bringing their problems, new hardware and software, and enthusiasm.

I have visited workshops of the above description run by other groups including the TRS80 group, and can vouch for the benefits that can be got from them.

Dealers or others who may want to have a presence at a Workshop by renting a table for commercial use, please contact me.
If you are able to talk on a particular topic, can demonstrate commercial programs or hardware for home or business use, or are just prepared to help with the logistics then again please contact me.

I hope that by organizing these get-togethers, some help will be given to those who want to start local groups, where they are not already in existence.

DATES FOR WORKSHOPS

September 25. Birmingham Poly. This is just off the Hagley Rd. See the latest Update for details of speakers/Demos.

October 23 London. Beaconsfield Rd School, Southall. Ian Trackman to talk on Structured Programming. Other details to be advised.

November 13 Nottingham University.(provisional)

COURSES

To date, two courses have been run by BASUG, the recent PASCAL course, and the Beginners Machine Code Course held in March. Both of these were highly rated by those who attended them, and both are to be repeated.
In addition three more courses are planned for the next year, Beginners and Advanced Basic, and Advanced Assembler.These are envisaged as weekend long courses. Subject to good turnouts on the Workshops, it may be possible to arrange one day courses on other topics, suggestions please. Again, it may be possible to run courses outside London, subject to speakers being available, any volunteers?

DATES FOR COURSES

October 22-24: PASCAL. This is a repeat of the highly successful July course. Cost £70 + accommodation.

December 4/5: Beginning Machine Code. With Ian Trackman, star of the BBC and Blue Chip Software. Approximate cost 15 pounds, to be held in London.
Please send for details of these courses, and on the Pascal course please HURRY.

EVENTS

As far as other events go, we will be exhibiting at the P.C.W. show on the 9-12th of September.
A new feature of our presence is that in conjunction with the TRS80 user group, we have hired the library of the Institute of Chartered Accountants at 44 London Wall, very close to the Barbican centre for the Saturday of the show.
The intention of this is to offer a place for members to relax and put their feet up after seeing what they want of the show. Those of you who experienced the Earls Court scrum will especially appreciate this. We are hoping to have bar facilities, and one or two Apple systems to

play with.
The next exhibition that we know of is the Northern Computer Fair at Bellevue Manchester on the 25-27 of November. With the Nottingham meeting and the Machine code course, resources will be very stretched. Free space is available through the A.C.C (Amateur Computer Club). Is anyone prepared to run a stall and carry the BASUG flag ?

more letters

County Down,
Ireland.

Dear Sir.
I would like to say how much I enjoyed reading Hardcore. I work mainly on my own, and have already found the magazine very useful.

I would like to pass on a little modification using a CCS Clock Card. This allows you to address the three programs on the card's Eprom under software control, and is very simple to implement. It entails removing the bridges at A8 and A9 on the card and running two wires from the pins to AN0 and AN1 on the Game Control.

The necessary Pokes and Truth Table are as follows:-

```
              A8  A9 :Routine.
Poke 49240,0 (     ):
Poke 49242,0 (0    0):Clock input.
Poke 49241,0 (     ):
Poke 49242,0 (0    1):Time string.
Poke 49240,0 (     ):
Poke 49243,0 (1    0):Screen display.
```

To save plugging and unplugging wires and Game control I have soldered a 16 pin wire wrap socket and two rows of Molex pins onto a small piece of Veroboard with one row of pins on either side of the socket, the tracks in the centre of the socket obviously being broken.

Yours sincerely,

Clifford Watt.

Wembley

Dear John
I really enjoyed the three-day course on Pascal and I'm sure that everyone else did too. I'll certainly be interested in other courses organised by BASUG.
Now there's a problem I hope you can answer.
My second disk drive recently went wrong and after I had it repaired I took m Apple in to check the drive, card and micro. The dealer told me he thought there was a fault in the Apple. We checked out the Apple using the Apple diagnostic disk and it appeared to be trouble free. But when the disk drive diagnostic speed controller was put into the same drive it would not run. Instead an error message "Overflow in Line 10" appeared on the screen.
When line 10 was listed it looked quite normal. But using the same disk, in the same drive, with a different Apple it ran normally.
The dealer also has at least one Apple in his shop with this same fault. He has also been in touch with Apple UK who appearently do not know the answer.
Wouldn't it be nice if BASUG could solve this problem and it would also be one up against Apple UK. And solve my problem and my dealers.

I hope you can help.

Yours sincerely,

Neville Ian Ash

(Ed. Neville. For my part I can't help much, but I can extend the problem further, by asking whether there is something wrong with the standard issue diagnostics disk. An Apple of mine was clearly faulty, and SBD's Master Diagnostics Disk laid the blame fair and square on a particular RAM chip - yet the regular diagnostics disk failed to detect any fault. Any reader's care to submit their experiences?)

PASCAL

"I was a seven-stone BASIC weakling, until I went on a BASUG PASCAL course !"
"I thought recursion was a criminal offence until I went on a BASUG Pascal course !"
"Since attending the recent PASCAL course, my wife has got rid of her lovers, I have had 3 promotions at work, I am a great attraction at parties, and I no longer lose sleep over personal freshness. "

Some, all or none of the above has been said by those who attended the last BASUG PASCAL course. BUT it may be said by those who attend the next. Don't miss the chance, send for more details now.

LUXURIOUS SETTING
FREEEE COFFEE
FREEEE LUNCH
2.5 days of ACTION!
ONLY 70 POUNDS

THE UCSD PASCAL COURSE

by Leo Crossfield

When one thinks of attending a course, one usually expects to spend part of the time learning and the majority of the time sitting, talking, and generally wasting time. This was not the case with the UCSD Pascal course. This course was a three day event which started on Satuday 17th July and finished on Monday 19th July, held at the Brunel campus 'Shoreditch', Runningmead.

The course structure was a series of workshops which were linked together by a number of well structured lectures. The lectures totalled nine hours over the three day course and were given by Drs Les Johnson and Mike Elstob from the department of Cybernetics at Brunel University. They gave an excellent series of lectures which alternated with workshops. The workshops functioned well with everybody working at their machines, while Les, Mike and myself made sure that no one was left unsure of the requirements and manner in which they should be attempting to solve the set problems. Constant guidance was available to everybody. The three helpers walked the floor constantly and guidance was given in a manner which allowed people to help each other to resolve their problems. Each workshop re-inforced the topics which had been covered in the previous lectures.

The aims of the course were to give each individual familiarity with structured programming, an understanding of the UCSD environment and to give confidence in developing their own programs. All these aims were achieved and more. The course was logically structured and had been designed with the experience of university teaching within the subject areas of Computer Science, Cybernetics and Philosophy. This gave each person the opportunity to have professional as well as personal tuition in methodology which was systematically linked to practical workshops. Everyone regardless of their backgrounds and past experience in programming, was catered for and by the end of the course had gained a great deal of confidence in both the design and the actual writing of their own programs.

The course was designed around a couple of simple programs which grew into a fairly complex algorithm including the use of nested procedures with flow of control loops, data input and validation, sort routines using arrays and disk file manipulation.

DAY ONE

People started arriving on the Friday night, from places as far as Manchester and Margate, in order to get ready for the Saturday morning start. The first day started with people setting up their equipment, some machines were shared. Dr Les Johnson started the course by introducing himself, Dr Mike Elstob and myself. The lecture structure was mainly led by Les Johnson with additional lecturing from Mike Elstob.

The lectures started with a highly detailed layman's guide to the nature of programming languages and how they are implemented on different machines. Compilers and interpreters were discussed, with regard to high level programming languages and a detailed discussion of the nature of the P-System. The P-System was shown to be a hybrid system in the sense that P-system compilers compile a language to a relevatively low level code (P-Code) which is later interpreted at run time to the individual machines executable code. Les also explained that this concept is a design philosophy to create a universal compatibility between different computer's architecture, the only fundamental difference being the P-code interpreter for each specific machine.

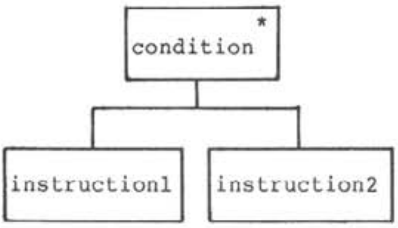
We covered the academic origins of Pascal and learnt that Professor Niklaus Wirth, within his lectures, attempted to standardise on a pseudo

language to demonstrate his ideas of block structured programming and data structures. Wirth later developed this teaching tool, which he had originally designed to allow programming to be taught as a structured and systematic discipline; a compiler was latter written for it. We next covered the concept of the UCSD System, and the later development of Pascal which gradually put our existing concept of Pascal into an historic context.

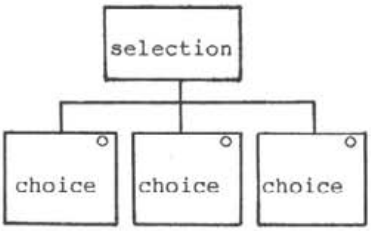
The lectures then turned to block structured program design with a heavy leaning towards the methods suggested by Michael Jackson. Three main structures "iteration" such as the 'WHILE DO' loop, "options" as in menu selection and finally "sequence" as in a set sequence of instructions. These three main structures are depicted below in the diagrams.

These structures were explained to be fundamentally tree-structures. We then looked at the nature of Pascal as a free formatted language. Indentation was discussed and the reasons for indentation was explained. We studied the concepts of bracketting and Venn diagrams and were shown how these concepts related to nested procedures and loops. We looked at the BEGIN and END statements, which were introduced as 'fat brackets', and the need for their use was understood due to the earlier lectures.

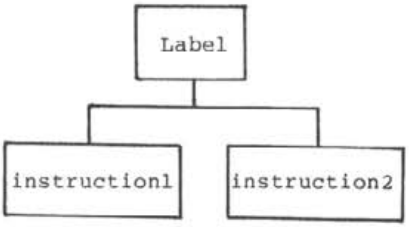
The first workshop was a critical, make or break, period. The whole course depended upon the confidence gained in this first workshop. This period was successful mainly due to the fact that Les Johnson had prepared an in depth tutorial which allowed everyone to walk step by step through the use of the UCSD Editor, Filer and the Main Menu.



Instructions are repeated until condition is met.



Selection will give access to one of the three choices.



Label consists of instruction1 followed by instruction2 but each is only executed once.

Everyone got on with their exercises without much need for assistance from the three helpers, or resorting to the many text books which people had brought with them.

All this was covered within the first morning, but had set the foundation stones for the entire course. Everyone had gained confidence with raw beginners sharing ideas with their more experienced colleagues.

Lunch was excellent and everyone returned eagerly to resume their studies. Throughout the course Les had one major problem and that was getting BASUG members away from their machines in order to have lunch and coffee breaks. Being used to university students, I felt that Les wasn't sure how to handle such conscientious students. He stood in the the doorway shouting instructions such as "IF not dead GOTO door ELSE stay where you are and don't move". Needless to say no one moved or even looked up, perhaps it was due to the lack of interest in the 'GOTO' statement.

The afternoon covered the logical development of a program which started off as a simple algorithm which displayed stars (*) on the screen. This program grew through specially designed stages to display a box structure, which was built up of stars and served to demonstrate the use of procedures with embeded and nested loop structures.

The day ended at 6-00 p.m. for the main class and I believe 2.00 a.m. for many of the keen ones. Rumour had it that arcade games were the main reason for such late night activity, as well as some discussions on the different problems and concepts which had been covered during the days activities.

DAY TWO

I arrived at 9.00 a.m., on a bright and sunny Sunday morning, ready to have a coffee and to prepare myself for the day ahead. I found a room full of self motivated people discussing problems with each other and working at their machines. The morning lecture started at 9.30 a.m. and program control was discussed in depth. Repeat Until, While Do, and For loops for repetition as well as the Case statement and the Then Else structure for controlling sequential instructions.

We covered the built-in string handling procedures to illustrate string manipulation. Throughout the course the concept of control and structure was amplified rather than the concepts of maths.

The course notes gently introduced the fundamental nature of set theory without anyone needing to realise that it is a mathematical concept. Sets were introduced because they are an important data structure in Pascal. We covered data structures and data types within the workshop which were intermittent with lectures; new concepts were introduced and put into practice. We covered data types such as strings, integers, reals, Boolean, char, arrays and sets.

The day's finale was designing and implementing a simple exchange sort procedure which illustrated the nature of nested procedures, local and global variables and an in depth use of arrays. This workshop supplemented the final sort by getting input from the keyboard, sorting the array and then outputing the result to the screen. A great deal of confidence had been gained within these first two days.

DAY THREE

The third day started with three very weary helpers attempting to look bright and alive. The atmosphere was relaxed and enjoyable, with the odd jovial remark from Jim Watson from Margate.

The final day's lecture started with a general introduction to the nature of recursion. The ancient game called the Towers of Hanoi was used to explain a standard example of recursive problem solving; each sub-problem essentially being the same problem as the main problem. This concept is comfortably handled by a block structured language. One can define the main problem within a procedure, which calls itself again and again until the total problem is solved. The Towers of Hanoi was not laboured within the lecture for the course notes go into greater depth. Indeed the whole course ran parallel with the course notes thus allowing the student, to go back over the notes to cover certain areas in more depth and at his own pace. This method obviously allows the student to retrace the steps of the course once he has left the studious environment.

The aim of the last morning was to analyse the game of 'Master Mind' which used all the skills learnt throughout the course. We were given a listing of the program, which gave each student a solid guide line for good program structure. We were given a step by step walk through Master Mind which allowed us to examine data validation and many of the other concepts that everyone had so far covered. We saw how one could define a Function and pass values to it with the function returning the product. Finally everyone was asked to critically access the main function in order to see how it could be written differently while still achieving the same goal; the goal being to compare four randomly generated integers to four integers which had been inputs from the keyboard.

The afternoon's aims were to introduce everybody to Records and Files. Mike Elstob took the floor and explained how to create mixed data types. Records in Pascal allow one to mix different types. For example, a structure could contain an element of integer, an element of string and a further element of integer. This structure is useful for manipulating a data structure such as a date within the program, i.e., '12 Feb 1982'. Records therefore allow very flexible manipulation of complex data structures, for one can easily increment the first field of integer without affecting the other two fields.

The next section of the lecture was based on a program called 'Data Capture'. This program served as a demonstration of external disk file structures. We were shown how to put data to disk, seek and then retrieve the data. Once everyone was used to these concepts we put them into practice with everyone essentially writing and manipulating their own simple data-base.

CONCLUSION

The course was a total succcess. I have gone in depth to explain the structure of the course in order to show the content and benefits which were gained by everyone. Everyone gained knowledge which could, theoretically, have been gained from one of the numerous books on Pascal, but how many of us ever manage to get further than a couple of chapters in most text books? Many of the people on the course admitted that they had, on numerous occasions, opened their Pascal tutors and closed them again without gaining very much for their endeavours.

With UCSD Pascal each new user has a number of problems to contend with. Firstly the UCSD systems environment which includes a powerful editor, the filer and compiler. Secondly one has to contend with a block structured language, which is fundamentally different to Apple Basic,

but once mastered will enhance the manner in which one uses Basic; making programs easier to design, understand and maintain.

The UCSD system is very different in concept to the standard Apple environment. The concepts of listing programs on the screen running programs, stopping the program to look at the variables, listing the program on a printer, all these concepts are different. The UCSD system is a

device driven environment, this essentially means that the console, disk drive and printer are all different devices with each device having its own specific volume number. For example, from the Filer you can transfer a file to the console, it will then appear on the screen, or you can transfer a file to another disk drive, or it can be transfered to the printer, to get a hard copy listing. The concept of a device driven environment is fundamental to the P-machine. All these points, plus the fact that you have a totally new syntax makes self tuition an enormous problem.

The course was excellent. I personally enjoyed the weekend immensely. The friendly atmosphere was created by so many different types of people all gathered together with a common interest. This after all is what club membership is all about.

The one thing that a course cannot offer is experience, but nonetheless it can arm people with the correct attitudes and tools to go alone to gain experience with confidence. What more can one ask of a course?

I would like to take this opportunity point to thank everyone that was involved in getting this course together. Special thanks to Drs Les Johnson and Mike Elstob for preparing the course. The lectures had obviously taken months to prepare, along with the course notes and demonstration programs. I would also like to thank the BASUG committee with special thanks to Francis Tao, the course events organiser. And final thanks must go to all the members that enrolled for the course. A jollier and more hard working bunch would be hard to find.

One last thing that I would like to add is :

"Hey Les, when is the next course?"

```
 0  REM  ********************
 1  REM  * M Y S T E R Y ?? *
 2  REM  *  STOLEN FROM      *
 3  REM  * C A L L  APPLE    *
 4  REM  *  AND MODIFIED     *
 5  REM  *     JULY 1982     *
 6  REM  ********************
 20  TEXT : HOME : DIM @(10):R = 1
     167
 30  CALL  - 1184: FOR M = 1 TO 80
     0: NEXT
 40  FOR I = 1 TO 10: READ @(I)
 50  FOR P = R TO R + 760 STEP 128
     : POKE P - 128,160: FOR M =
     1 TO 50: NEXT M: POKE P,@(I)
     : NEXT P
 60 R = R + 1: NEXT I
 70  VTAB 23: END
 80  DATA   66,110,65,110,83,110,8
     5,110,71,110,
```

MACHINE CODE COURSE

by Bob Raikes

This two-day course on Beginning Machine Code was a follow up by Ian Trackman to his one evening talk on the subject to the Herts & North London BASUG group. One of the first things to be apparent was that the whole subject is much easier to understand when a machine is at hand to try programming on.

Over the two days we progressed from the basics of "What is binary ?" to writing programs which would clear the hi-res graphics screen quickly, and restore an Applesoft program accidentally erased by the use of FP or NEW. On the way, the mysteries of using the DOS Toolkit Assembler and the Monitor's inbuilt routines were revealed.

While those who went on the course were not all beginners, everyone benefited from the course, and the main question at the end of the weekend was "When is the next one?".

EDUCATION COLUMN

by Norah Arnold

THE EDUCATION GROUP

I have learned from Frances Teo that some people who joined BASUG at Slough may be interested in joining the Education SIG. If you have already written, then your letter is probably still in the pipeline and will be reaching me shortly. If you would like to join but haven't done anything about it, write to me via the BASUG Box Number.

How about a bit more feedback from those who joined a few months ago? Now is the time to have a think about the last educational year and tell us what you found most useful in your work, what problems you met and what you hope to do in the future.

TERRAPIN LOGO, CHAPTER II.

As we have now reached the end of one educational year and are preparing to begin another, it is a good time for me to look back and assess the effect that Terrapin Logo had upon the eight to nine year olds I began using it with in the Spring term. Without exception the children enjoyed their experience with Logo, although some obviously took their interest much further than others.

On reflection, certain landmarks can be identified in the children's approach to programming in Logo. The day when two or three of them came to the conclusion that it would be worthwhile trying to find out why some turtle-graphics programs had more bugs and seemed very difficult compared to others, was probably the first of these. Their conclusion, that if they consistently used angles of 180, 90, 60, 45 or 30 degrees life was a great deal easier, led them to work out the following reference table:-

No. of degrees.	No. of turns necessary to get back to the place you started from.
360	1
180	2
120	3
90	4
72	5
60	6
45	8
40	9
36	10
30	12
24	15
20	18

J.R. Systems
presents.....

User Services.....

*Consultancy
*Software development
*Insurance

Coming Attractions....

* The Building Suite
£55
- Builders' package

*Slider
£35
- Photograph indexing

For Details.....

Write to J.R.S.
2 Rye Close
Harpenden
Herts.

Tel. 05827 5100

From this point some of the brightest children progressed by leaps and bounds. Although they did not take the table further, it made them recognise what an interesting number 360 was, after all 'nearly everything divided into it!'.

The other magic number was of course 90 degrees. Some children got so stuck with it that they became proficient in writing bug-free graphics programs using only 90 degrees and a few other angles.
Another landmark was conquering the circle. After a wet dinner-hour one boy was able to pronounce, "I haven't spoken to anyone for half an hour but I've done it. It's just got to be REPEAT 360 (FORWARD 1 RIGHT 1)." He didn't look back after that.

Neither was the Logo confined to Maths lessons. The extensive list-handling capabilities available led to it creeping quite naturally into English and Science lessons and forming a large part of a highly successful Art project.
By the way, Apple Logo and Terrapin Logo are both based on M.I.T. Logo and are almost identical.

COURSEWARE WRITER, a review.

A problem that I experienced when using Aristotle's Apple to create 'one-off' revision quizzes for older students has been solved for me by the use of the Teacher's Toolkit (Courseware Writer) available from Wida Software. The version I found most useful is upper-case only, which confines its use to text-based subjects for older students such as History, Sociology, Economics etc., although lower-case versions for various foreign languages are available.

The CREATION program which is used to construct the question sequence, and the TESTER which enables the question sequence to be run by the student, are the workhorses of the suite. As with all test construction programs, thorough preparation definitely pays dividends and the use of these programs then becomes easy and straightforward.

The fact that you are allowed up to eight lines for each question makes the Toolkit program handy for 'wordy' subjects where it is difficult to make questions short without them also becoming ambiguous. The flexibility of the answer area is another good point. Answers can take up to eight lines, and these can be arranged in various ways. One could have eight alternative correct answers but I found it a good idea to give one or two alternative correct answers followed by a positive reinforcement comment (to follow a correct response by the user) and a negative reinforcement comment (to encourage after an incorrect response). A little ingenuity in thinking up varied and interesting responses is repaid by the fact that the user of the test does not get annoyed by the same response to his answers coming up over and over again.
The tests I created with this program have worked consistently, but one or two minor irritations cropped up when using the other programs which make up the suite. I sorted these problems out fairly rapidly but someone less familiar with the Apple might have greater difficulty. However, the gentlemen of Wida Software would no doubt help to exterminate any bugs which may have crept into their programs.

BASUG TOOL KIT NUMBER 2
GET HEX

by Cliff Wootton

This tool kit routine will allow you to read the keyboard and test the value entered, converting it into hex. The entry values of the 6502 registers are irrelevant (i.e they will be scrubbed anyway). The first example is a single hex digit input routine. This is illustrated in Listing A.

```
GETHEX JSR RDKEY
       CMP #KZERO
       BCC GETHEX
       CMP #KTEN
       BCC OKDONE
       CMP #KA
       BCC GETHEX
       CMP #KG
       BCS GETHEX
       CLC
       ADC #$09
OKDONE AND #$0F
       RTS
```

Listing A Single Hex Digit Routine

Of course, you will more often want to get a two digit hex value. This will fill an entire byte which should be very simple (isn't it all?). The next routine that is listed will accomplish this by calling the previous GETHEX routine twice and combining the two results. This is illustrated in Listing B.

```
GETBYTE JSR GETHEX
        ASL A
        ASL A
        ASL A
        ASL A
        STA TEMP
        JSR GETHEX
        ORA TEMP
        RTS
```

Listing B Get Two Hex Digits

When getting data from the keyboard in this way you often need to display the characters accepted, as they are entered. It is by far the simplest solution to output them after they have been tested for validity. If they are printed as soon as the character is entered, invalid characters must be back-spaced over and deleted. This is an easy enough task in itself but for a subroutine to work in the most logical (and structured) manner it must have only one entrance and one exit. To instruct a subroutine not to back-space on entry but to back-space each time it error loops is not so simple without leaving and reentering the subroutine at some point other than the start. This breaks the rules of structured programming which may not be everything but often "they're all we've got". The easiest solution is to print the contents of the accumulator on return from the GETHEX routine.

Listing C shows how to get hex digits and print the results.

```
GPRHEX JSR GETHEX
       PHA
       JSR PRHEX
       PLA
       RTS
```

Listing C Get And Print Hex Digit

Depending on the routine you use to print the contents of the accumulator you may or may not need the PHA and PLA instructions. To implement this modification to the double-digit-get-hex routine, this PHA-PRHEX-PLA sequence is required after both calls to the GETHEX routine. Listing D shows how the print function can be integrated in this way.

```
GPBYTE JSR GETHEX
       PHA
       JSR PRHEX
       PLA
       ASL A
       ASL A
       ASL A
       ASL A
       STA TEMP
       JSR GETHEX
       PHA
       JSR PRHEX
       PLA
       ORA TEMP
       RTS
```

Listing D Get And Print A Hex Byte

A more compact version might be as shown in Listing E.

```
GPBYT2 JSR GPRHEX
       ASL A
       ASL A
       ASL A
       ASL A
       STA TEMP
       JSR GPRHEX
       ORA TEMP
       RTS
```

Listing E Get And Print A Hex Byte 2

A short explanation of the variables and syntax of these listings may be in order. I do all of my machine code assembly with the APPLE TED assembler. This was a forerunner of the APPLE TOOL KIT. Where I have called a routine that has not specifically been described then it is from the Apple monitor. You may wish to use your own routines if these are not suitable for your application. Load immediate is signified by the # character which should be read as a hash sign.The immediate constants such as KZERO and KTEN are the numeric key range values. KZERO equates to $B0 while KTEN equates to $BA. The constants KA and KG are the range checks for the alphabetic keys. The KA value equates to $C1 while KG equates to $C7. The variable called TEMP can be any location zero page or otherwise as you wish. Give these routines a try in your spare time. If you find a shorter way to get the same result please write it up for the journal. There are no prizes however. That is unless any one wants to donate an APPLE ///. If so I would be more than pleased to 'look after' it until I can decide on a winner.

ANNUAL GENERAL MEETING

Minutes of The AGM of BASUG, held at the Old School, Park St, St Albans at 2.45 pm on Sunday 18 July 1982.

The meeting was opened by John Sharp, BASUG secretary. Apologies were presented for the absence of the Chairman, Frank Kay.
It was proposed that Mike Preston be elected Chairman for the purposes of the meeting. This having been carried, the Chairman pointed out that there was not a quorum and proposed that the meeting carry on as a normal meeting until such time as a quorum was present.

1. It was proposed by John Martin, and seconded Jim Panks that the minutes of the last AGM be taken as read. There being no objections, this was carried.

2. John Sharp then gave the report of the committee.

The membership stood at approximately 1100, over three times the number a year ago. During the year BASUG had attended the PCW Show in September, the Holland Computer Club show in Utrecht in November, the ALCC Fair at the North London Polytechnic, and the Computer Fair at Earl's Court also in April, with the APPLE 82 show at Slough to round off the year.

The name of the group is becoming known throughout the world. Contact has been made with the IAC via the European Director Wolfgang Dederichs, meetings with him at Utrecht, and in June before and after his visit to the IAC AGM in Boston.

The workload has been very high, and some jobs have not been done as well as they might have been due to lack of time. One neglected area has been the relationship with APPLE(UK). Although we have set up a relationship, meetings and contacts have been infrequent. We helped them put out a marketing questionnaire, which they will now analyse and send us the findings. More could be done if the effort was put in, i.e. Brian Reynolds has agreed that information can be put in the boxes with the APPLEs sold.
We have provided a number of services to members. Most letters have been answered satisfactorily. The group has PRESTEL facilities available which need to be developed in the coming year.

The issues of HARDCORE are maintaining the same standard, and there have been many original articles which have not appeared in other magazines. Both HARDCORE and UPDATE have meant a great deal of work by a small group of people, often into the early hours of the morning (in fact with one recent case, until 5 o'clock in the morning.

Merchandising has become a major service to the group, with a steady increase throughout the year. Initially it took in software and 'other items', with John Rogers sending out software and David Bolton the discs, etc. In January the committee took the decision to pay a secretary to send out both as one order. This has now grown to be part time job for three mornings a week.
Letters are coming into the PO BOX in a steady stream, in the region of 200 a week. It takes hours just to sort them. At one time there was a peak after HARDCORE had been despatched, dying down when the next one was due. Now there is a steady flow.
John Rogers has maintained an increasing stock of programs, almost certainly the largest of a user group in the country. A start has been made on documenting the programs, and an announcement on progress will be made soon. There have been problems with copies but there are few delays now, average turnover being within a week unless there is a specific copying problem, or stocks run out.
At the beginning of April, the committee resolved to retain John Sharp as a part time paid consultant, not only to ensure the continuance of the group, but also in recognition of the effort already put in. Further developments in this direction are inevitable. The letters and general work has increased, and it is essential that someone be available all day to answer questions chase advertising, supplies, etc.

HARDCORE still continues to fill 48 pages, and apart from providing a much needed exchange of ideas it usually manages to provide some new insight into the workings of the APPLE every issue. Advertising has dropped off of late, because there has not been time to chase it. The committee sees that its successor will have to make moves to put HARDCORE on a firmer more commercial footing as has been done with the merchandising.
COURSES and events have been a 'Cinderella' service during the past year. Events organiser Eddie Payne resigned in November leaving a vacuum until Fran Teo was co-opted. Despite this setback there have been a number of successes. As stated previously, BASUG has been prominent at all the major shows.

The Machine Code course was very successful despite a struggle to make up the numbers. The same seems to be true of the PASCAL course.

The LIBRARY has been another area where response has not been forthcoming. This is probably due to the high cost of postage. The

exchange of newsletters/magazines with other groups has been poorer than expected. It is felt that perhaps we are too professional with HARDCORE.

3. The Treasurer pointed out that he was not in a position to present the report (see below).

4. The meeting not yet being quorate there could be no election for committee members.

5. Discussion then began on a number of topics.

a) John Wellsman asked whether as the TRS80 User Group have seminars and workshop weekends, which were well supported, could we not do the same. It was pointed out that the problems with course events organisation over the past year had not allowed such things to happen. Bob Raikes as nominee for the position thought something could be done along those lines.

Mike Preston thought the lateness of disseminating information was one problem, that a three-day PASCAL course was too costly, that the need was for one-day meetings, etc. The feedback from the show suggested that the majority of members wanted HARDCORE and the software and they were not too interested in courses/meetings. There would also have to be some changes with any new APPLES to add to the present ones.

b) John Martin asked how financially viable the club was, and if so how much we could offload as paid work. The treasurer stated that there was no doubt of the financial viability. Prices were set to make a profit. The magazine does not make a profit. The subscription basically pays for the £2000 approximate cost (incl postage) of each issue. Advertising needs chasing, contributions are not paid for, but there is a positive cash flow, which could be improved provided there was someone to put the effort in.
Norah Arnold saw the need for more publicity.
Mike Preston said that eventually a full time employee was required. We were not at the CALL APPLE level, but we were moving in that direction.
At this stage in the proceedings, more members arrived, the meeting was quorate.

VOTING FOR NEW COMMITTEE

Three members were proposed for Chairman-
Norah Arnold proposed by John Rodger and seconded by Bob Raikes
- John Wellsman proposed by John Martin and seconded by Jim Panks
- John Martin proposed by Fran Teo and seconded by John Wellsman.

The rest of the committee was nominated and seconded en bloc; nominated by Mike Preston and seconded by Cliff Wooton. The names were as follows:-

Secretary	John Sharp
Treasurer	Fran Teo
Membership Secretary	Jim Panks
Hardcore committee members	Tony Williams David Bolton
Software librarian	John Rogers
Information officer	John Rodger
Courses/events officer	Bob Raikes

Norah Arnold was voted Chairman and the rest of the committee was unanimously accepted.

Mike Preston then proposed that John Wellsman, and John Martin be elected also, this was seconded by Fran Teo and carried unanimously.

Discussion then ensued regarding the position of local groups. As there were only a small number at present, the committee could coopt a member from each, for a meeting when necessary. It was proposed by John Newell that it would be up to the groups to nominate themselves for cooption to such meetings, and by Cliff Wootton that a minimum of 20 members be in the groups to be eligible.
John Martin proposed that the AGM sanction extending the practice of engaging part time support as and when needed at the discretion of the committee. This was seconded by Vivian Parkins, and carried by unanimously.
The secretary put forward a suggestion by the outgoing committee that a work evening be held each Tuesday when there was not a local group meeting at Park Street. This was seen as a means not only of spreading the load, but also of involving more members in the running of the club.

Discussion on the accounts led to John Martin proposing and John Wellsman seconding that:-
The AGM agree to formally pass the authority for approval of the accounts to the newly formed committee and that the outgoing treasurer produce such accounts as soon as possible, with a view to the committee approving them and passing the active balance to the new treasurer.
This was carried. John Martin volunteered to audit the accounts.

Membership subscription was then discussed, with the following being proposed by John Martin and seconded by Bob Raikes:-
i) The subscription excluding the joining fee be raised.
ii) That it be not more than £15
iii) The final amount be agreed by the committee

All points were carried.
Discussion also ensued regarding the need for reminder of renewal, and of the possibility of starting a rolling membership (i.e. being annual from date of joining). This was referred to the committee.

Cliff Wootton suggested that publicity could be obtained by getting into the MICRO magazine bibliography. This was referred to John Rodger for action.
There was also need for an information sheet for new members.

There being no other business, Mike Preston proposed a vote of thanks to everyone who had helped during the past year and closed the meeting at 5.10pm.

mickie

A special purpose language for multiple choice tests and quizzes, questionnaires and Computer aided learning (CAL) – **£50** + VAT

★ MICKIE programs can be written by people with no previous experience of programming. Many people have neither the inclination nor the aptitude to master general purpose languages such as BASIC or Pascal.
★ MICKIE can be used by people who have never used or even seen a computer before. This is demonstrated by MICKIE's success with hospital patients.
★ MICKIE is written in a simple to use, easy to remember format, designed specifically for the first time user who does not want to know more than is absolutely necessary about computers and computer languages.
★ Originally developed for medical history taking. MICKIE has been evaluated in hospitals, schools and commerce. It saves time and provides comprehensive, legible and structured records.
★ The original MICKIE was developed at the National Physical Laboratory by the late Dr. Christopher Evans (well known as the author of 'The Mighty Micro' and 'The Making of the Micro').

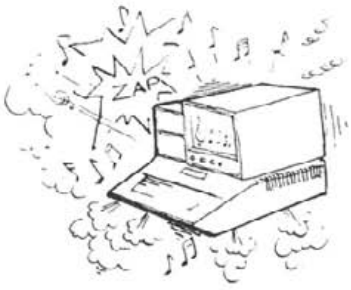
Full Apple implementation from:

SYSTEMICS LIMITED
21-23, The Bridge, Harrow, Middlesex HA3 5AG.
Tel: 01-863 0079

MICKIE is a trademark of the National Physical Laboratory

ZAPPLE

SOUND EFFECTS & MUSIC BOARD



- Built-in software means easy to use from Basic.
- Software included to easily create "Sound Tables".
- ALLOWS SOUND WITHOUT SLOWING GRAPHICS
- Built-in amplifier gives volume control of all Apple sound output.
- 3 voices + noise expandable to 9 voices

Volume control / amplifier also available separately. Send £18.

We also do 1-off and small run PCB and assembly work.

We do custom-made software, too. Ask us.

Send £65.

Tel: Leighton Buzzard (0525) 370621

MEEKROSE

Meekrose Ltd
85 Waterdell
Leighton Buzzard
Bedfordshire
LU7 8PL

AMATEUR RADIO - RTTY THE VERY EASY WAY

by Derek Turner G8 LDR

This article is aimed mainly at radio amateurs and others who are already familiar with radio teletype (RTTY from now on), and judging from the letters to BASUG and the interest shown at Apple 82 there are many who would welcome a description of the system I use.

RTTY is a system which enables two machines to communicate text by means of radio waves. Radio amateurs discovered in the early sixties that they could use ex GPO Telex machines to communicate world wide in conditions previously possible using only Morse code. In the last few years a renewal of interest in RTTY has taken place with the advent of the micro. This article describes one such implementation.

Basically the Apple is connected via the games port to a modem which in turn is connected to a radio transmitter receiver. A real time clock provides once per second interrupts which provide the heartbeat for the intelligence routines. The program is written in machine code which carries out the fast bits and Integer BASIC which carries out control and string handling.

The normal display consists of two independently scrolling windows separated horizontally by a status line about three quarters of the way down the screen. Incoming text is displayed without line feeds in the top screen while text for sending may be set up simultaneously in the bottom window. The receive store defaults to a size of 16k which constitutes a good few hours monitoring. The send side of the program is particularly powerful and includes many string handling features designed to allow the user to type ahead while the machine sends out the often repeated phrases or sayings that are heard all over the amateur bands.

Rather than become verbose I am merely going to list part of the instructions and hope that the initiated will understand the finer points. The most powerful of these is possibly the set of routines which allow the machine to operate in unattended answerback mode. In this the machine if called will answer with the minimum of text and say that it is ready to receive a message. If a message for the station calling has been stored by the user then this will be transmitted. In any case if the station calling is known to the system then he will be greeted by name.

INSTRUCTIONS

CONT A tx/rx switch
CONT O use second tx store
CONT T repeat tx
CONT R force letters shift
CONT L run tuning routine

ESC delimit keyword for message string

CONTROL E SUBCOMMANDS
QRZ capture callsign of station calling
and update message strings
NME input name of other operator
BGN reset contact start time
SER reset contest serial number
ANS put machine in answerback mode
MES update the five messages for specific stations

CONTROL B SUBCOMMANDS

CLR clear tx or rx store
GET load text file to send
PUT save text on disc
DIR read names of text files on disc
SYS display memory status
AUT set software autostart mode
REV review receive store , rx text
normal , tx text inverse
WRT dump receive store to printer
rx text upper case , tx lower
BUF change tx rx partition of memory

CONTROL S SUBCOMMANDS

SPD set baud rate
CDE set baudot ascii or morse
MDE send each character or wait for
space then send word
FIL set fill character
WID adjust send width
ALF suppress most punctuation characters
TXT display BASIC'S stored messages
LF suppress receive line feeds
UNS force letters shift after space

SPECIAL STRINGS (set up from BASIC)

TIM TIME 1905 GMT
Q 599 001 AT 1905 GMT IN ZL28D WATFORD QSL?
G6XXX DE G8LDR K K K
CALL G8LDR DE G6XXX
OVER BACK TO YOU BILL . G6XXX DE G8LDR PSE K K K
BYE TNX FER QSO BILL . G6XXX DE G8LDR SKSK
K PSE K K K
CQ CQ CQ DE G8LDR etc.

That's it so far. I am still working on more refinements such as a selcal system for recognising specific items on commercial RTTY stations and have been having odd thoughts about error correcting systems such as AMTOR; and as well as this a system for automatically scanning for a station calling CQ.

The most interesting aspect of the whole system for me has been operating the answerback system. There is no feeling so strange as that when digging the garden and hearing the tones coming up from the shack and it's your station talking to someone by itself!

Sunningdale, Berkshire.

Gentlemen,

I have run into rather a peculiar situation which has puzzled me, my son and all that I have asked so far. I ran 'STOCK MARKET GAME' which crashes if you respond to line 18's question with a 'Y' (Branch Err)
So I tried to list 12000 - nothing !! Then I listed the whole program - yes everything lists OK, well almost, lines 555, 560, 565, 570 all are repeated !
I took a hard copy and then deleted these lines then re-entered them, result still the same. Anyone got any thoughts ????

My daughter who is now 10, has been found to be dyslexic and I am wondering if any members have had any experience with educational software in this context? It is her recognition of the written word that needs to be practised. She seems to have little trouble with her figures. We are waiting for her written assessment from the Centre which should enable us to choose her remedial instruction.

Yours sincerely,

Peter Trinder

(Ed. There seems to be an observable correlation between dyslexia and compatibility with computers, so your daughter can count herself lucky to have such a father. However, for more practical help about specific packages to help reading, you could do worse than drop a line to BASUG member Bryan Spielman of Sinta Software, who is working on the kind of package you need - and, of course, our very own Norah Arnold has quite a bit of experience in this area and will be glad to give advice).

NICELISTER HEADINGS

by M J Parrott

I was impressed by the program 'Nicelister' in the latest issue of 'Hard Core'; it certainly improves printer listings considerably. However, I like to put a heading on my listings using an IMMEDIATE mode program and the use of the & and the PR# commands before NICELISTER precludes this. Hence I added a little more to the program (see the enclosed hexadecimal dump) which then gives the user the ability to print such a heading.

With the modified version of NICELISTER the Applesoft program is put into RAM, NICELISTER is BRUN and the slot number, printer width, and title, (if required) are entered after the appropriate prompt has been made. A slot number of 0 merely causes a listing on the screen. One thing I noted is that printers which issue a line-feed with a carriage return are best used by entering a printer width value of one less than the full width.

To modify the original NICELISTER, BLOAD it and enter the hexadecimal values as given in the dump, change the JUMP at $8000 to go to $8280 (i.e. 8000:4C 80 82) and BSAVE it as LISTER,A$8000,L$30A.

HEXADECIMAL DUMP OF ADDITION TO 'NICELISTER'

```
A8259.8309

8259- C5 CE D4 C5 D2 A0 D0
8260- D2 C9 CE D4 C5 D2 A0 00
8268- D3 CC CF D4 A0 00 D7 C9
8270- C4 D4 C8 A0 00 D4 C9 D4
8278- CC C5 A0 00 00 00 00 00
8280- A5 36 48 A5 37 48 20 58
8288- FC A0 82 A9 59 20 3A DB
8290- A0 82 A9 68 20 3A DB 20
8298- 2C D5 20 01 83 20 F8 E6
82A0- 8A D0 09 A5 36 48 A5 37
82A8- 48 4C B5 82 D8 18 A9 00
82B0- 48 8A 69 C0 48 A0 82 A9
82B8- 59 20 3A DB A0 82 A9 6E
82C0- 20 3A DB 20 2C D5 20 01
82C8- 83 20 F8 E6 86 E0 A0 82
82D0- A9 59 20 3A DB A0 82 A9
82D8- 75 20 3A DB 20 FB DA 20
82E0- 2C D5 68 85 37 68 85 36
82E8- A9 00 A0 02 20 3A DB 20
82F0- FB DA 20 65 81 20 FB DA
82F8- 68 85 37 68 85 36 4C D0
8300- 03 A9 00 85 B8 A9 02 85
8308- B9 60
```

FRENCH KEYBOARD FOR APPLE II

by Tony Williams

Hard on the heels of Apple's long awaited announcement of foreign language versions of the Apple III comes a software product from Microsource which enables the trusty Apple II to display French characters on the screen. This package has been a long time in the making not because of any inherent difficulty but because of the restrictions presented by the Apple keyboard - there are just not enough keys to go round - and the large number of extra characters required by French.

Microsource, which seems to concentrate on devising packages for off-beat applications of microcomputers, developed this routine for Wida Software, the educational software specialists who produce computer aided courseware, mainly for the teaching of foreign languages. One year ago Microsource wrote a successful lower case and German loader which Wida has incorporated in its bumper 'Apfeldeutsch' package and other German teaching programs.

The main stumbling block with French is the sheer number of extra characters - 16 in all - and there is no way to implement them on the Apple without resorting to a double keying sequence. Wida Software ruled out the use of hardware solutions, since its customers are almost exclusively language teachers and learners with only limited access to Apples. They insist on using the machine 'as is'.

After long discussions it was decided to access the French keys in the following manner (as illustrated) which at first seems highly intricate but which in fact needs about ten minutes familiarization, comparable to that required by a dedicated French typewriter.

If you want to access non-standard characters the Apple normally steers you towards use of the ESC, CTRL or CTRL-SHIFT conventions. However, in order to retain normal CTRL operations as far as possible, Microsource opted for the double dead key SHIFT-N solution instead. This means that on wanting to access, say, a grave è, you first signal the coming change by typing SHIFT-N, then signal which change by SHIFT-> followed by the letter on which the grave is to be placed, i.e. e and voilà, it emerges as è. If a circumflex is needed, again signal the change by SHIFT-N, state which change by SHIFT-^ and then the letter required, say a, producing â.

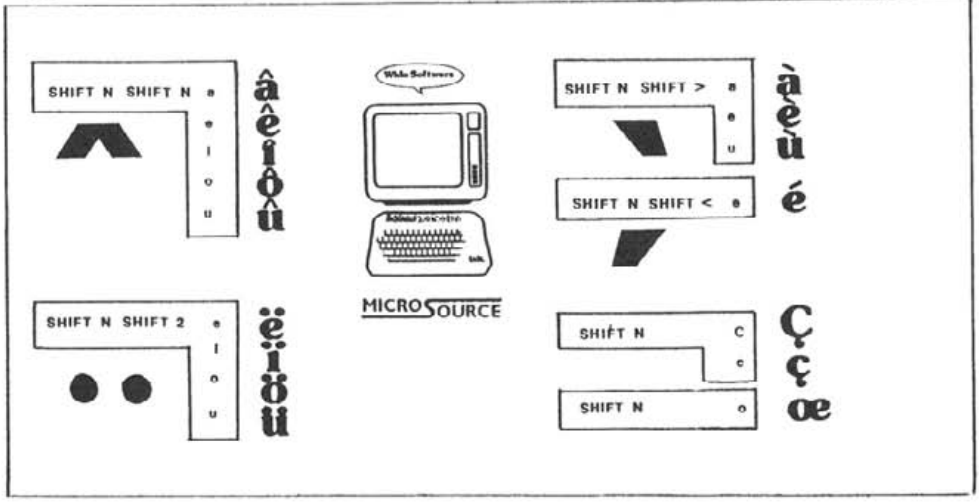
As far as possible the secondary dead key gives a visual clue to the sign it will produce, so that SHIFT-N gives ^ (the circumflex), SHIFT-2 (") gives a diaeresis, SHIFT-< suggests an acute and SHIFT-> suggests a grave accent.

Luckily French only uses these letters in lower case, except for Ç cedilla. This is accessed by SHIFT-N alone, followed by C or c. Ça va?

Another concession to the Apple's original configuration is reflected in the QWERTY keyboard. While it would have been entirely possible to reconfigure it to AZERTY (and

FRENCH KEYBOARD FOR THE APPLE II

1. BOOT THE WIDA SOFTWARE FRENCH DISK
2. ENTER CALL-2048: CALL 2072 (DIRECT OR INDIRECT)
3. PRESS ESC TO ENTER LOWER CASE MODE
4. PRESS THE FOLLOWING KEY SEQUENCES:



ZWERTY in the case of German) it was decided that it was not worth the hassle. This concession also meant that there was no need for press-on keytops indicating the new keyboard. The keyboard remains essentially unaltered and a prop-up illustration (as below) is a sufficient memory prod. This sketch is pasted to the disk sleeve (an ingenious idea this) and also repeated in the package's documentation.

The Microsource/Wida Software lowercase and French character generator is written in machine code and after booting the special disk it is activated in your BASIC program by CALL 2048 (which accesses lower case) and CALL 2072 which specifies the language required, in this case French. German would be CALL 2069.

The switch between upper and lower case is toggled by the ESC key. From lower case one press of the ESC key gives one upper case letter, a double press locks into upper case. One practical problem with conventions of this kind is knowing at any time just which case one is operating in (when an input is expected, for instance). This package dispenses with ugly status lines and exploits the cursor prompt: a steady flashing underline indicates upper case; a rapidly flashing underline lower case.

On the debit side, I have to say that the lower case's usurpation of ESC as the lower case toggle is irritating in that, naturally enough, ESC can no longer be used to get into I-J-K-M cursor control. Microsource gets round this by using CTRL-Z instead, which does work, but I don't like it. Fortunately the package is compatible both with the Program Line Editor and with Super Editor (Boot PLE first, then RUN Super Editor then RUN the Hello program on the French disk which places the character set in memory.)

Using these facilities enables one to largely forget the clumsiness of CTRL-Z cursor control.

It would have been welcome to retain the neat Applesoft HOME:TEXT:INVERSE: NORMAL instructions, but this seems to have proved impossible. The first two have been replaced by CALL 2057 and CALL 2054. INVERSE is activated by CONTROL-I and NORMAL by CTRL-N - which makes them extremely easy to rememer and use. FLASH is not available at all. Clear to the end of the window is now CALL 2060 and Clear to end of line is CALL 2063. When converting an existing BASIC program to Lower-case (French or German version) extensive use can be made of the Global Find and Replace feature of Super Editor to make these changes (before CALL 2048 has been made)

At the moment the French routine has been implemented in Wida Software's authoring system, its 'Teacher's Toolkit', and in special French teaching courseware which it is developing. It has been designed for use within a BASIC program and not for word processing applications.

THE HERTS. GROUP

NEXT MEETING SEPTEMBER 7TH.

A demonstration of the Alpha-Syntauri computer music synthesizer.

This instrument is a high performance synthesizer featuring a velocity-sensitive keyboard, a programmable eight-way keyboard split, sound on sound recording, plus extensive sounds definition software.

Reviewing the alpha-Syntauri in Byte magazine, December 1981, Steve Levine and Bill Mauchly wrote:-

"The system is more than just an Apple peripheral; it is a musical instrument in its own right. Its price and performance clearly place it beside commercial synthesizers made by Moog, Oberheim, Arp, Yamaha and Sequential Circuits."

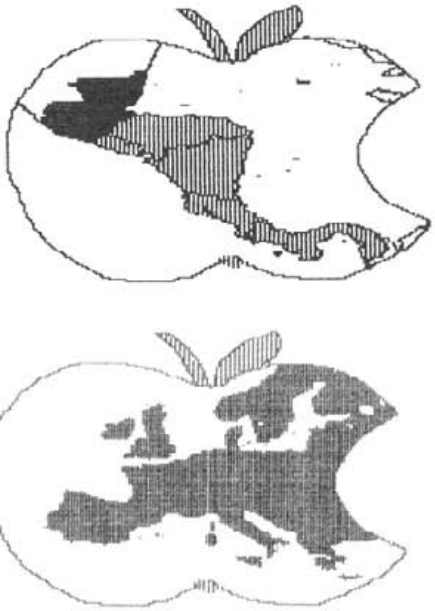
Don't miss this meeting if you are interested in computer music.

OCTOBER 5TH. PROGRAM GENERATORS.

It is hoped that this will include a demo of "The Last One" and C.O.R.P.

Herts. group meetings are held at:-

The Old School,
Park Street,
St. Albans.



APPLEWRITER II

A review of the latest word processor from Apple.

By Jim Panks and Richard Teed

I have heard people say that the new Applewriter II is the best thing to come from Apple. While I would not go quite that far I can say that it is superior to the old Applewriter 1.1. It is written by Paul Lutus and distributed by Apple Computers under the Special Delivery Software label. It is available from all good Apple dealers and it costs in the region of #75 plus the usual 15% V.A.T.

The documentation is a very well- written Operating Manual in the usual Apple fashion. It consists of 106 pages divided into three main chapters, under the titles Text Editor, Printing and Word Processing Language. The last title is interesting. Applewriter II has a powerful language of it's own called WPL and it is used in the program; all the help menus and screens are written using it. The Operating Manual is written to a high standard and it would be nice to find all operating manuals written to the same high standard.

Applewriter II can be used in the normal manner, that is to say with 40 columns. It also has the facility to make use of a Lower Case Chip, and to my surprise it can be used with the Super 'R' Terminal 80-Column card.

On booting up in the normal way, you are asked if a Lower Case Chip is present. After answering this, you are presented with the Editor Menu. This contains a copyright notice and a list of the most frequently used control characters; you are asked to make a selection from this menu. At this stage you can load a file directly into the Editor or you can go into the Help Menu which consists of a tutorial on the facilities offered by Applewriter II. A quick comment on the Help menu is in order here. If at any time you get stuck whilst using Applewriter, you can go into the Help menu without touching the text already entered in the Editor. This is a nice facility and it beats grabbing for the Operating Manual every few minutes.

There are three main menus, the Editor Menu, Additional Functions Menu and the DOS Menu; they are all obtained by simple commands. The Editor Menu has already been explained and the next menu is the Additional Functions Menu. This is obtained at any time by pressing control Q. This menu allows the loading of print, tab and glossary files, it also allows the display of carriage returns and use of the Shift Key Option. From this menu Applewriter 1.1 files can be transferred to Applewriter II files, which is a very good idea considering the number of users. This aspect is displayed throughout the program and the documentation, and from my point of view is a strong selling point.

The DOS Menu allows the manipulation of files without the destruction of files already in memory; it allows you to initialize, delete, rename, lock, unlock and catalog the disk of your choice.

When you first enter the Editor proper you are presented with an inverse bar at the top of the screen; this is optional but helpful. The bar is full of information; it includes the current mode of the cursor, the amount of memory left, the length of text entered, the present position of the cursor, the current position in the paragraph and the name of the current file in memory.

The cursor is a flashing block with an underline in the opposite colour, and it changes when the current cursor entry mode is changed, so that when entering capitals you get a '^' in the centre of the cursor. When in the cursor move mode, it changes to a flashing '@'.

In the cursor move mode the I,J,K,L keys are used for single movements and the E,S,D,X keys for up,down 12 lines and left,right 24 characters or one word. The arrow keys are used for deleting and retrieving characters.

The control keys B and E put the cursor at the beginning or the end of the file, the key C gives automatic case change , and K locks uppercase.

There are facilities to delete, move, retrieve and find characters, letters, words and paragraphs. It is possible to save parts of text to disc, and to retrieve parts. It can also display other text files from disc with the file in memory untouched. This is ideal when you want to find a piece of text from another file and enter it into the present file.

Tabulation is looked after very well and it is possible to have 'tab files'. This saves loading tabs manually every time you need a specific tab set.

Applewriter II allows easy footnotes, underlining, subscript and superscript although you must have a printer that will obey the commands. To help users a group of special characters have been implemented, these include underline, reverse slash and brackets {}.

Printing the file is easy. You can set up print files and load them from disc, or you can alter the default file, so that the most used print file will be available every time.

The file lets you alter the following parameters;
Margins(Left,Paragraph,Top,Bottom,Right,)-Page number - Printed lines - Page Interval - Line Interval - Single Page - Printer Slot - Carriage Return - Printer Mode (Left,Fill,Centre,Right Justify) - Heading (Top and Bottom Line) - New Page - Continue Printing.

All the above parameters can be embedded in the text or set using the WPL. Other commands that can be used in the printing mode include the stopping of printing and the sounding of a bell followed by a message on the screen -- "Any one for tea - press return when last biscuit eaten." It is very useful when using single sheets or doing mailing shots. You can make it print a hundred then stop for tea or to let it cool down.

A useful addition is the ability to put relative values in the text, so that you can make pieces of text move around the paper. All you do is put .lm+5 and it will move the margin right 5 spaces from the previously set left margin.
(.lm+5)
(.rm-5)
To put the text back you just put .lm-5 and it lets the left margin return to it's proper place; it is very nice and works well.
(.lm-5)
(.rm+5)

How many times have you wished that you could look at two parts of a file at the same time ? NEVER ? Well, it is one of those split-screen routines, (just like Visicalc ?). You can work in either of the sections and it makes comparing text a doddle !

Now I will try to explain the WPL. It is fairly simple to grasp and uses a language that does not use line numbers, and if saved on disk it can be made into real structured sub-routines; that is to say, you can make a number of routines that will be capable of being strung together to form a larger more sophisticated program.

On the Applewriter II disk are several WPL routines that allow, for instance, the automatic printing of address labels, the counting of the number of occasions that a word is used, the automatic formatting of a letter with personal greetings on it. Altogether it is a very useful idea and will I think be of great assistance to serious users.

If your spelling is as bad as mine, it is possible to use Goodspell to check the spelling in Applewriter II files.

Well I hope my article does justice to Applewriter II because I believe that without buying 80 column cards,Z80 cards or very expensive software , you will not find better value ...!

Richard Teed has helped in the writing of this article and his comments on his experience using Applewriter II over the last two months follow!

The replace command has a difficult to use command structure, with special characters for wild cards and returns, if these special characters are needed in the search string as the characters themselves then a different set must be used, for example you can have!-

< for the delimiter
= for any length
> for return and
? for wild card

If you want to use a "?" for example in the search string then you must choose from another set of replace characters such as!-

# for the delimiter
$ for any length
% for return and
& for wild card

A much easier syntax which many other editors use is something like!-

/ or <escape> for the delimiter
= for any length
<return> for return
control A to interpret the next character not as a special character
control Z for wild card
control N for any character except the next following the control N.

The above I feel would make the search and replace commands a lot easier to use.

The manual is with out doubt very well written and easy to follow but it really does need an index.

My final complaint is with the Word Processing Language! Paul Lutus clearly designed this program for editing blocks of text but it is so good that it can be used for creating assembly language files and other line orientated files, what is therefore needed of WPL is the ability for the macros to work on a line by line basis which they are unable to do at the moment. By

allowing All Applewriter II commands to be included in macros (that means cursor moving commands) this word processor / text editor would be one of the most powerful available for any machine, and that includes larger computers, not just micros.

I am grateful to Mass Micros of Welwyn Garden City for allowing me to borrow Applewriter II for this review.

Dear Mr. Sharp

I recently attended the three day Pascal course (at Brunel University) organised by B.A.S.U.G.

Although I am a virtual newcomer to computers in every way (I have had an Apple II for four months) I found the course extremely stimulating and enlightening.
Dr. Les Johnson, leading the course, provided an intensive 'hands-on' approach to Pascal. Briefly this incorporated lectures on programming languages, the P-system, structured programming, control structures in Pascal, procedures, functions, recursion and data. The lectures were well supported by practical workshop sessions.

With programming experience I undoubtedly would have gleaned much more from the course. However, thanks to the advice I received on books, to the moral support of the tutors and to the revelation that Pascal is a very manageable language, I feel now that learning Pascal is actually possible and well under way.

I would strongly recommend such a course to any B.A.S.U.G. member and should like to thank Dr. Johnson for his help and support.

I should add that the food at Brunel is also up to the standard of the teaching!
As an afterthought, is there anyone in the Berkshire area, conversant in Pascal, who would be willing to discuss the occasional problems which arrise? If so please phone me at CROWTHORNE 5191, or contact me at the above address.

Yours sincerely,

Roger M. Sherwin

Selangor, Malaysia

Dear Sir,

I have just purchased an Apple II computer and I have a few problems which I think you might be able to help.

Firstly, about the 'RESET' key. How can I inhibit the 'RESET' key and at the end of the program, enable it OR create an autostart program such that whenever the 'RESET' key is depressed, it will not 'break' or stop the current program but will load and run a program called HELLO.

Secondly, I would like to point out that the Copyright Act in my country is not strict, therefore pirate copies of programs are quite easily obtained. At the moment I am freelancing for a local company here which is dealing with APPLE computers. I find that some of my routines are stolen and used by the company for their benefit. I get paid about 40% each time my package is sold. I found out that the company makes copies of the packages and selling them without paying my share. As computer sales in this country is slow, it is the only company dealing with customised software that suit the Malaysian condition.

Can you please tell me how to change the address of the Address Field or the Data Field of a disk thus avoiding copies being made. The Address Field normally start with the bytes $D5/$AA/$96. If any of this bytes were changed, DOS would not be able to locate that particular Address Field causing an error. Any alternative method to avoid copying would be appreciated.
Thank you for your kind cooperation and I hope in due time, there would be an APPLE club here too.

Yours sincerely,

Derek Wee

(Ed. There are programs available which allow you to protect your own programs which run under DOS 3.3. The protections include the ability to cause the program to self run if the user attempts to list or modify the program or press 'RESET'. Also the disks cannot be copied with normal copy programs. An example of this sort of program is PROTECTOR III from SBD software. Other companies such as INNOVATIVE DESIGN or QUALITY SOFTWARE protect their own programs and may be able to advise you on protection schemes.

A dedicated pirate will copy your programs no matter what protection schemes are used, but the above measures will put off most people.)

L.A.U.G.H.S.

Leicester Apple Users Group for Help and Support

If you own or have access to an APPLE computer you need to be a member of **L.A.U.G.H.S.**

The benefits of membership are:-

1) Automatic membership of the British Apple System Users Group (BASUG).

2) This includes 6 copies of "HARDCORE" the club magazine per year.

3) Access to the software libraries of both the group and BASUG.

4) Access to the Library of both the group and BASUG.

5) Membership of the International Apple Core through BASUG.

6) Discounts available from both local dealers and the main BASUG group.

7) Courses on various topics.

8) Access to fellow Apple sufferers for the Help and Support you need.

How to join

Come to one of our monthly meetings at the "WINSTANLY ARMS", (See map over) on the first Wednesday in each month, or phone Hazel Bown the club secretary on Leicester 875253.

AFFILIATED TO B.A.S.U.G.

TIME MACHINE II

by Ian Pawson

The Time Machine]I[is a real time clock/calendar card for the Apple]I[. It provides time and date information with programmable interrupts of 1/1024,1,60, and 3600 seconds. The card has a rechargeable battery that will power the clock chip for up to 10 weeks. It is trickle charged when the Apple is switched on.

The clock provides six date formats and seven time formats, all software selectable. The onboard 2k EPROM occupies the $Cn00 RAM space and does not conflict with the $C800 space. The EPROM contains programs to read the time and date from simple BASIC statements, allows the user to send commands to the clock, and interrupt routines to display the date and time on the top line of the screen, and to write the date and time into an Applesoft string.

After reset or power up, the default date and 24 hour time format is: DD-MMM-19YY HH:MM:SS It is somewhat unusual for an American product to display the date in UK format. This can be changed, if required, with the Format command to one of seven time and six date formats.

To display the date and time on the top line of the screen just type PR#n:PRINT"A1":PR#0<RETURN> (The card is in slot #n.) The other interrupt routine included writes the date and time into an Applesoft string called 'TIME$'. This string must have been created first, and must have a maximum of 31 spaces. The date and time can then be read with a PRINT TIME$ statement.

There are three commands that can be used from a BASIC program, INPUT" ";CLOCK$ will put day of the week, date, and time into the string CLOCK$. INPUT "DATE";DA$ will put the date into the string DA$. INPUT "TIME";TIME$ will put the time in TIME$. These commands must be preceded by IN#n:PR#n to access the clock card.

The manual contains examples, in both BASIC and machine code, for using the various functions of the Time Machine. A disk of software is also included, and this contains programs to set the clock, demonstrate the interrupt facilities, and the various display formats.

By far the most useful, however, is a modified DOS. This date stamps all files saved or accessed. It limits program names to 19 characters, the last 11 contain the date. It is not necessary to type the date information when loading or running a program as the DOS only looks at the first 19 characters. When transfering files using FID or Super Disk Copy, using the 'wildcard' function after the file name will enable the file to be located without the date information being entered.

All in all, a very useful device. I cannot imagine files without date stamps. They make it very easy to find the latest version of a program, or the date the backup copy was made, or the last time a text file was accessed.

The Time Machine]I[is available in the UK from Protocol Computer Products.

SNAFU TIME AGAIN

by Simon Brown

Yes, it's horrendous snafu time again!
Having recently submitted a group of assembler source code listings, I have found a major problem with 'Applesoft Variable Dump'. It is that having dumped the variables, it restarts the program! So here are the patches (not very elegant, but they do work).

1) Remove line 2 (STXTPT EQU $D&97)
2) Replace line 80 (JMP STXTPT)with a RTS.
3) Remove from the accompanying description the part about resetting TXTPTR and the associated problem.

As it now stands it won't restart the program but after it dumps the variables it causes a 'SYNTAX ERROR'! The fix is shown below but this increases its length to $EA and so, if used in Page 3, it may (if it hasn't already) overwrite some of DOS.

1) Insert immediately after the CLD in line 20. Thus:-

```
LDA POINTER    ;SAVE TXTPTR
PHA
LDA POINTER+1
PHA
```

2) Insert immediately before the altered RTS (as above) this:-

```
PLA            ;RESTORE
STA POINTER+1  ;TXTPTR
PLA
STA POINTER
```

Now everything is OK and it can now be used within a program. Hope this doesn't cause too much trouble.

BEGINNERS PAGES

STRINGS

Soon after starting to program in BASIC one realises there are two types of variables - numbers and 'strings' of characters. Numbers can be assigned to string variables, but they are as a set of characters just like words, and do not have any meaning as regards the position of the number in the sequence as occurs with numerical variables.

Applesoft has a number of commands to handle strings. Integer Basic does not have as many. There are a number of ways of getting round the problems. Some of these are described in the book APPLE II USERS' GUIDE, others are outlined and used in the program "THE INFINITE NUMBER OF MONKEYS" by Bruce Togazzini on the library disk 25.

In the following I do not propose to give the full command structure in each case. The syntax is in the APPLESOFT manual or books like "APPLE II USERS' GUIDE".

With these commands we can add strings together:-

```
10 A$ = "ABCD"
20 B$ = "EFGH"
30 C$ = A$ + B$
40 PRINT C$
```

and C$ will be printed as ABCDEFGH.

This process can also be useful in the addition of "illegal" characters to strings which have been input. APPLESOFT will remove leading spaces if you try to type them in, and in some cases you need them in another part of your program. This can be accomplished by the following:-

```
10 INPUT A$
20 B$= CHR$(32) + A$
30 C$= " " + A$
```

BOTH lines 20 and 30 undertake this manipulation in different ways. Other characters such as "," can be added in the same way.

We can remove them from the end of a string

```
10 A$ = "ABCD"
20 B$ = LEFT$ (A$,2)
30 C$ = RIGHT$(A$,3)
40 PRINT B$
50 PRINT C$
```

This will then print AB and then BCD, as the values of B$ and C$.

We can combine these operations to move one or more characters from one end of a word to another:-

```
10 P$ = "ABCD"
20 Q$ = LEFT$(A$,1)
30 R$ = P$ + Q$
40 S$ = RIGHT$(R$,4)
```

In practice this would be done in one expression

```
40 S$= RIGHT$(P$,3) + LEFT$(P$,1)
```

Normally, we would not necessarily know the length of the string, as it could vary, or be the result of an INPUT statement. There is another command to allow you to find the length of the string A$ - LEN(A$). Now we can shift from one end of any length string:-

```
10 INPUT X$
20 Y$ = LEFT$(X$,1)
30 Z$ = X$ + Y$
40 P = LEN(Z$) - 1
50 W$ = RIGHT$(Z$,P)
```

Where might one use this? One place would be the case as follows.
Suppose we had a list of names and we wished to sort them. We would normally do this on the basis of the first letter of the surname. This means we would store the surname as the first word followed by the forename. However, as we know from when BASUG's membership database was written like that, some people object when they receive letters addressed to BLOGGS FRED. One way to get around this is to switch the strings around. We then make use of what is known as a delimiter - in this case the space between the words, to know where to make the split. There are two parts to this. The first is to find the space and the second is to do the switch.

Finding the space is accomplished by means of the command MID$. In practice this is done as follows:-

```
10 INPUT A$
20 FOR N = 1 TO LEN(A$)
30 IF MID$(A$,N,1) = CHR$(32) THEN 100
40 NEXT N
50 PRINT "NO SPACE!!!" : END
100 B$= LEFT$(A$,N-1)
110 C$= RIGHT$(A$,LEN(A$)-N)
120 D$= C$ + " " + B$
130 PRINT D$
```

These various functions are vital for handling strings. Some BASICs contain another command called INSTR. This allows a target string to be searched for in another string. This is especially

useful in the case of a question and answer system, where one wants to trace a particular answer in a host of variations e.g. the string "RABBIT" out of such possibilities as "A RABBIT" or "RABBIT" or "RABBITS". This is where APPLESOFT begins to look a bit long in the tooth. Perhaps someone would like to write in with a simulation of the INSTR function, from the string handling functions already present.

MID$ can be a very useful in another context. Sometimes it can be a problem storing many variables in arrays. The variables can be stored in strings. The months of the year for example:-

```
10          YEAR$          =
"JANFEBMARAPRMAYJUNJULAUGSEPOCTNOV
DEC"

20 INPUT "WHICH MONTH DO YOU WANT
(1-12)" ; A

30 MONTH$ = MID$(YEAR$, ((A-1) + 1 ), 3)

40 PRINT "THE MONTH IS :- " ; MONTH$
```

This is only of use when the substrings are of the same length. There is a famous method of inserting machine code into memory by using this technique. Perhaps someone would care to write about the LAM routines for doing this.

YOUR EAMON PROBLEMS SOLVED

by John Martin

A number of queries have been received about EAMON disks, and we are now able to deal with the problems raised.

1. The Eamon Master disk (E1) is required not only to run " Beginner's Cave " (on the same disk) but also to run ANY other EAMON adventure (disks E2-E6)
2. Some copies of the Master Disk are now known to contain a faulty set-up program. You can ascertain whether this applies to your disk by LOADing and LISTing the program " The Wonderful World of EAMON ". If you spot nonsense within this program, return it for replacement.
3. Otherwise, proceed as follows :-

a) Remove write protect tab from disk E1
b) Run Master Disk program " Set up for 32K ", if you have 48K RAM.
c) RUN " The Wonderful World of EAMON "
d) Follow the instructions on the screen
e) When prompted, either press 'C' key for Beginner's Cave on the same disk, or replace with another adventure disk and THEN press 'C' key.

Happy Adventuring !!

We've opened up the whole country.

SPECIAL PRINTER PRICES Aug/Sept All Models Floppy Disks £12 per box

MASS MICROS

- *WE SUPPLY NATIONWIDE*
- *WE INSTALL NATIONWIDE*
- *MAINTAINED NATIONWIDE*
- *BEST PRICES NATIONWIDE*
- *LEASING SCHEMES*

Our unique service allows you to buy anywhere in the UK with the confidence and knowledge that the equipment we supply is at the right price and supported.
Our experience of putting together the correct system for you ensures full satisfaction.

- COMPLETE RANGE OF SOFTWARE, PRINTERS AND PERIPHERALS.
- 12 MONTHS NEXT DAY SERVICE INCLUDING PARTS, LABOUR AND TRAVEL TO YOUR SITE.

- ASK FOR SERVICING QUOTES ON NEW AND EXISTING EQUIPMENT

MASS MICROS

Wellson House, Brownfields,
Welwyn Garden City, Herts.
Tel: Welwyn Garden (07073) 31436/7
Telex 298641

Please send for FREE information pack on Apple ☐ CBM ☐ Superbrain ☐
Printers ☐ Maintenance ☐
Name
Company
Address
Tel No.

Input Fix

Author: Simon Brown

When an input is typed in via the keyboard, once the RETURN is pressed the rest of the line is erased. This is OK for most situations, but if, say, you have presented a form on the screen to indicate what answer is required when, and want to step other parts of the form on the same line being erased, it becomes extremly tedious resetting the text window each time an answer is required.

My solution goes part way to solving this problem. It is two short machine code routines that are placed either side of an input routine (High or low level) and can be used in immediate and deferred execution modes. The first stores the line of screen text that is on the same line as the requested input and then the input routine is executed. The second half then takes over. This section starts at the rightmost position of the relevent line, going backwards replacing the old line on top of the blanks caused by the input, until it finds on the screen a character that is neither a space or the same as was there before, then it stops.

The reason it is only a partial solution is that if the line that is inputed goes off the right edge of the screen, the blanking effect occurs on the next line down, so the restoration routine is ineffective (although no harm is done to the display, the usual blanking effect occurs)

```
*300.33F

0300- A5 28 85 F0 A5 29 85 F1
0308- A0 27 B1 F0 99 D0 02 88
0310- 10 F8 A2 00 20 75 FD A0
0318- 02 8A 91 69 C8 A9 00 91
0320- 69 C8 A9 02 91 69 20 39
0328- D5 A0 27 B1 F0 C9 A0 F0
0330- 05 D9 D0 02 D0 08 B9 D0
0338- 02 91 F0 88 10 ED 60 4A
```


Wida Software

Apfeldeutsch £120.00
CAL course in German. Beginners to post O'level. Textbook; Workbook; 6 Audio Cassettes; 7 Apple disks of fully interactive teaching and testing routines. Apple only (20% discount for schools)

Teacher's Toolkit £25.00
Courseware writer
Starter pack for building up library of CAL courseware. No knowledge of computing needed. Master and Student disks include Tester, Creation, Editor, Directory.
Available in French, German and Dutch versions (£30.00)

Hebrew II
Hebrew only £60.00
Hebrew and English £100.00
Mini-word processor for Hebrew. Scrolls from right to left. Includes vowels. Needs Apple II, 1 disk drive and Silentype or graphics printer. Perfect for teaching students Hebrew spelling and typing.

Apple Pilot £84.00
Apple's very own authoring language. Combine text, sound and graphics on screen to create impressive interactive and demonstration packages. Needs two drives

Master Maths £90.00
Complete O'level course in Maths by LCL. Uses Apple graphics and sound. Disks available separately (£20.00)

Shape Manager £46.00
The easy way to build up shapes and use them in your program. Does for shapes what a word processor does for words. Sinta Software.

Storyboard £10.00
Reconstruct page of text on screen (with clues, next word, etc). Ideal for language learning, memory reinforcement. Very addictive.

Mickie £50.00
Computerized interviewing system. Devised by the late Christopher Evans originally for medical history-taking. Provides comprehensive structured records. By Systemics.

Timetabling
Book (Hutchinsons) (no VAT) £10.95

Timetabling Software £30.00
by John Gray. Stay posted for new developments.

Terms - Please add 15% VAT to all except book orders. Cheque P.O with order or Visa/Access. Send 9" X 6" s.a.e. for free catalogue.

WIDA SOFTWARE
2 Nicholas Gardens · London W5 5HY
Tel. (01) 567-6941 · (062) 882-5206

```
 1 CURPOS     EQU       $20
 2 LINEADD    EQU       $F0
 3 SAVELINE   EQU       $2D0
 4 ;
 5            ORG       $300
 6 ;
 7            LDA       CURPOS          ;MOVE CURRENT
 8            STA       LINEADD         ;CURSOR ADD
 9            LDA       CURPOS+1        ;TO
10            STA       LINEADD+1       ;LINEADD
11 ;
12            LDY       £$27
13 SAVE       LDA       (LINEADD),Y     ;SAVE LINE
14            STA       SAVELINE,Y      ;IN TOP OF
15            DEY                       ;INPUT BUFFER
16            BPL       SAVE
17 ;
18 ;
19            LDX       £$00            ;INPUT ROUTINE
20            JSR       $FD75
21            LDY       £$2             ;THIS IS IAN
22            TXA                       ; TRACKMANS
23            STA       ($69),Y         ;FIRST INPUT
24            INY                       ;ANYTHING
25            LDA       £$00
26            STA       ($69),Y
27            INY
28            LDA       £$2
29            STA       ($69),Y
30            JSR       $D539           ;END OF INPUT
31 ;
32 ;
33            LDY       £$27            ;START FROM RIGHT
34 RESTORE    LDA       (LINEADD),Y
35            CMP       £$A0            ;IS THERE SPACE CHR ON SCREEN?
36            BEQ       STORECHR
37            CMP       SAVELINE,Y      ;IS SAME CHR IN SAVED AND ON SCREEN
38            BNE       END             ;IF NEITHER THEN END
39 STORECHR   LDA       SAVELINE,Y      ;PUT SAVED CHR
40            STA       (LINEADD),Y     ;BACK ON SCREEN
41            DEY
42            BPL       RESTORE         ;END OF LINE?
43 ;
44 END        RTS
```

```
100  REM ************************
110  REM *  HOW TO PUT QUOTES   *
120  REM *  IN A PROGRAM LINE   *
130  REM *     ENTERED FROM     *
140  REM *     AN EXEC FILE     *
150  REM *                      *
160  REM *    BY R.D. PURVES    *
170  REM ************************
175 X = 63000
180  PRINT :D$ =  CHR$ (4)
190  PRINT D$"OPEN EXECDEMO"
200  PRINT D$"WRITE EXECDEMO"
202  PRINT X"HOME"
204  PRINT X + 1"LIST"X + 2
210  PRINT X + 2"?CHR$(34)";
215  PRINT  CHR$ (34)"HELLO";
220  PRINT  CHR$ (34)"CHR$(34)"
230  PRINT "RUN"X
240  PRINT D$"CLOSE"
250  PRINT D$"EXEC EXECDEMO"
```

Verbatim® Datalife™

disks for the performance of a lifetime

NOW disks, with the five year warranty

MINIDISKS

Verbatim, the world's leading producer of minidisks, makes the widest range of minidisks available for word processing and data processing applications. Intent on maintaining this position of leadership, we constantly strive to make our minidisks even better.

Each Verbatim Datalife™ minidisk is equipped with a hub reinforcing ring to aid in registration, reduce errors and give better alignment repeatability. Our dual-sided, double density Datalife™ minidisks offer users higher storage capacity. Our new plastic box makes our minidisks easier to use, safer and more convenient to store. Our testing standards go beyond the industry standard because we insist on Verbatim being the industry standard of excellence.

Introducing

CLEANING DISKS

The Verbatim Datalife™ Head Cleaning Disk is a convenient, reliable, safe means of removing up to 90 percent of debris contaminating magnetic recording heads used in computer and word processing systems.

The Verbatim Datalife™ Head Cleaning Kit consists of a durable, reusable, Lexan® jacket and presaturated, disposable cleaning disks.

The Verbatim approach is superior to and less expensive than other cleaning disks presently available.

The kit, available in 8" and 5¼" sizes, differs from those already available in the marketplace in the following ways:

- The Verbatim cleaning disks are disposable. This approach allows a fresh clean disk to be used each time. No accumulated debris from previous cleanings comes in contact with the heads.
- Each cleaning disk is presaturated with a precise amount of cleaning solution. Pre-saturation eliminates the inconvenience associated with splashing solvents onto a disk.

Head Cleaning kits 5 1/4 in only
$6 incl P&P
Replacement cleaning disks
$11 per box of 10 incl P&P
Available from BASUG

SPECIAL OFFER TO B.A.S.U.G. MEMBERS ONLY

10 x MINIDISKETTES **£19·50** INCLUSIVE OF V.A.T. AND POST.
I ENCLOSE A CHEQUE MADE PAYABLE TO B.A.S.U.G. CHEQUE No ..... PLEASE FORWARD TO ME BY RETURN AT THE ENCLOSED ADDRESS.

NAME .....

ADDRESS .....

..... POST CODE

London N.W.3.

Dear David,

Graham Rubens (Letters - June 1982) asks for a DOS patch to stop DOS from opening an empty text-file when a named file does not already exist on disk.

May I suggest an easier method which does not involve any DOS patching. Enable error trapping then VERIFY the file. If it doesn't exist, the error trap will be "sprung" with the "FILE NOT FOUND" error and can be used to skip the unwanted "OPEN" command. Here is an example :-

```
.
.
.
100 ON ERROR GOTO 1000
110 PRINT D$ "VERIFY TEST.DATA"
120 POKE 216,0
130 PRINT D$ "OPEN TEST.DATA"
.
.
.
1000 POKE 216,0
1010 IF PEEK (222) = 6 THEN PRINT "FILE NOT ON
DISK"
.
.
```

Graham also asks why Apple Writer cannot cope with two disk controllers. The answer is that Apple Writer's internal commands for transferring between its two modules are "BRUN TEDITOR,D1" and "BRUN PRINTER,D1". "D" ($C4) can be can changed to "S" ($D3) at $12C8 (in Teditor) and $102C (in Printer).

The third problem, which I can only answer here in outline, is how to put Teditor and Printer into memory at the same time without using a Ramcard. Each module starts at $803 and ends just before $1900, so taking up roughly $1100 bytes (i.e. 3.75K each). The text buffer is located between $1900 and DOS's buffers at $9600, taking up approximately $7C00 bytes - about 19K.

Two ways of putting both modules into memory are a) to store them both under DOS and then copy the appropriate one down to $803 as required and b) to relocate the second module at the top of memory and call it there.

The first method, which follows the Ramcard idea, needs twice 3.75K and would reduce the text buffer to about 11.5K. The second way needs only 3.75K, leaving 15.25K. Extra memory could be released by reducing MAXFILES to 1.

Unfortunately, Apple Writer does not make automatic internal adjustments to compensate for changes in HIMEM and MAXFILES and these would need to be altered manually in the code. Furthermore, the second method would involve off-setting all absolute addresses in the code so that the re-located module would run properly at its new location - not too difficult by using a symbolic disassembler (such as SYMDIS) and then re-assembling the source code.

If there is sufficient interest from BASUG members (letters to Hard Core and not to me please), I will try to work out the methods in detail. However, it should be mentioned at this stage that Apple Writer is a copyrighted program and the source code of the patched programs cannot be published or put into the BASUG library.

Ian Trackman

AUTOCRAT COMPUTERS

DAISYWHEEL
PRINTER/TYPEWRITER
OLIVETTI PRAXIS 30
ONLY £ 475

MEMOREX
HUB REINFORCED FLOPPY DISKS
5¼" BOX OF TEN
ONLY £ 14.95

VAT EXTRA

264 Preston Road, Harrow, Middlesex,
TEL: 01-908 3636 HA3 0PY.

Salisbury

Dear Fran

... As for the Pascal course itself, I would like to say how well it was presented. I cannot say that I enjoyed it - it was hard going and we were not there to enjoy ourselves - but the subject was covered extremely competently. The two lecturers and Leo Crossfield as demonstrator were all extremely 'user friendly'! I rated the course a great success and I hope it is repeated for the benefit of others in future. May I wish BASUG every success in this respect. Again, many thanks for putting on the course and thanks to all who made it possible.

Yours sincerely

Clive Duggleby
BASUG Member [redacted]

Bowes Park

Dear Sirs,

Could you help me with the following query/problem. I have just installed a 16k memory expansion card into my Apple system (ribbon cabled to 3E memory chip!). Now DOS 3.3 still sits in the usual position and Integer Basic goes into the card on start up. I wish to put DOS there and use the extra memory for programming.
1. Is there an easy way to do this?
2. Is there a program in the library to do this?
3. On Page 15 of Hard Core for February 1982 there is an article on relocated DOS in "Call Apple" for July/August edition and Nov/December, both 1981. Is this in the literature library?

...I would also appreciate it if you oculd help me with the following. I have bought two items, secondhand:
1. Programmer's Aid No. 1 Chip.
2. Microchess No 2.0 Disk.

Can anyone help me with details of operation for either, especially Item No 1, as no details of the contents are known by myself.

Yours faithfully

Tom L Beckham

Basug Member: [redacted]

Bexley

Dear John,

I write concerning a problem I have had with my Apple Assembler. I am starting to hack my way through the world of machine code programming by trying to write a prograsm and learning commands as I go along. I was, however, dismayed to find that the Assembler would not accept the Opcode 'ASL'. After ensuring that I had not slipped up, I assumed it to be one of the 'bugs' referred to in Ian Trackman's recent review, though I have since discovered that 'LSR' and 'ROR' are also rejected by the assembler with a BAD Expression error. I should be grateful if you would advise me whether this is an error on my disk, or a genuine bug in the Apple Assembler/Editor, and if the latter is so, recommend an assembler that works!

I would also like to offer the following tip/suggestion which may be of interest.
When preparing text for printing it is often desirable to include certain words in single quotes. On the Apple screen the apostrophe fills this role very well, but with some printers, the Epsons, for example, the apostrophe curves to the left like a comma, so ideally the open quote should curve to the right. This character is available and is effectively the lowercase equivalent of '@'. It can be generated from 'Apple Writer' as follows. (The actual keypresses are shown within brackets).

1. Enter '@' (Shift P)
2. Enter cursor control mode and put cursor in front of '@' (ESC ESC J)
3. Enter case change mode and advance cursor over '@' (CRTL-C K)
4. Return to text mode.

You should now have the '@' displayed inverse, but it will be printed as an open quote by the printer.

Yours sincerely

Simon Brown

(Colin Richardson writes:
I assume from your enquiry about ASL, LSR, ROR, etc, that you are using the DOS Toolkit Assembler, and it seems as if you may have been used to the mini-assembler where ASL etc is accepted. With the Toolkit Assembler you must specify that the ASL is to be performed on the accumulator, so the correct format is ASL A (and similarly LSR A, ROR A, ROL A) with the A in the operand field (i.e. there must be a space between LSR and A - some assemblers use LSRA with no space. Since the mini-assembler assumes that all numbers are hexadecimal an instruction such as LSR A would be interpreted as LSR $A, which is why you must use LSR etc.
I hope this solves your problem).

Saturday July 24th 1982.

The Editor,
Hardcore,
B.A.S.U.G,Watford.

Dear David,

Apathy ?

Come, come, old boy, this down in the mouth stuff will not do. It never pays to wallow in self pity and "negative thinking" the like of which is creeping into Hardcore just lately. Look at things-- positively !

It is a well known fact among organisations that participation by a membership is at worst non existent, and at best minimal. This is more usually so in nationwide memberships of diverse interests, such as the mixed bunch of "video screen gawpers" that makes BASUG, to which I am proud to belong.

BASUG'Sdiversity is both its strength and its weakness.

On the one hand, BASUG has grown by appealing to a wide spectrum of computer interested individuals. That Hardcore is distributed and achieves a response, if only in letters such as this would mean it is read. It may even used to solve a problem, or referenced to find a product. Any one of these uses would justify the existence of BASUG and the publication of Hardcore ! That most of us use it for A.L.L. of these reasons is a bonus to the member !

On the otherhand, because we are so diverse, participation in individual projects will be sparse. Meetings will not necessarily be successful if you measure success by numbers in attendance rather than the benefit those who turned up got out of it.

So, David, let's not run ourselves down, our computer widows and widowers will do that for us!

Yours sincerely,

Jim Watson.

ADD AUTO REPEAT ON YOUR KEYS

from a recent IAC communication.

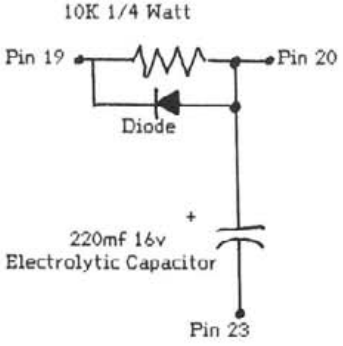
If you wish to modify your APPLE II to allow for automatic repeat - by holding a key down for more than a half a second - the following modification can be carried out. As usual this voids your warranty note.

a) Parts required - 10K 1/4 watt resistor, 220 mf 16V low leakage electrolytic capacitor and a 1N4148 diode (1N914 or 1N4150 can also be used).

b) Construction, assemble the three components outside the APPLE. Lay the diode parallel to the resistor body and wrap the diode leads around the resistor leads. Cut off the extra and solder.

Looking into the APPLE, locate the pins on the piggyback keyboard encoder assembly directly under the keyboard. Find pins 19 and 20 (counting from the left if you are looking from the keyboard to the back). Now, observing the diode polarity, connect the + (striped) side to pin 19 and the other end of the diode to pin 20.

Take the electrolytic capacitor and connect to pin 20 (+) and pin 23 of the same assembly.



WANTED!!!!

Has anybody got a copy of NIBBLE volume 3 number 1? I will pay #5 for a copy!.
Contact Martin Rogers on [illegible]

Wanted

Can someone lend a copy of NIBBLE Vol 3 no 1 to

Peter Baron

Telephone [illegible]

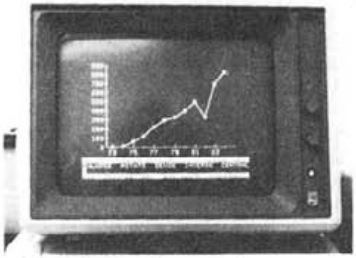
lower case chip + SHIFT KEY MOD

- Adds the full lower case set to your APPLE II
- Complete with clip on SHIFT KEY modification
- Lower case letters have true descenders
- Compatible with most word processing packages
- Available in most European languages
- Comprehensive documentation covers installation & software

Requires Revision 7 (and upward) APPLE II
PRICE £34.00 + V.A.T.

Visiplot Driver Routines

If you have Visiplot, a graphics printer and one of the following – Grappler, Digitek Printmaster, Pkaso, then our intelligent printer and Driver Routine is a must.



- Send direct from Screen to graphics printer – allowing image to be enlarged, rotated, printed black/white, white/black, centralised etc.
- Simple menu selection enables you to send graphics screen image directly to printer.

Our Visiplot intelligent driver routines allows you to take full advantage of your printer card capabilities.

£19.95 + V.A.T.

THE ROUTINE MACHINE

By Peter Meyer

At last an easy-to-use way of putting the POWER and SPEED of machine language routines in YOUR OWN APPLESOFT PROGRAMS!

ROUTINE MACHINE does all the work for you – no knowledge of machine language programming required. Simply choose the routine you need from an every-growing library, and Routine Machine will effortlessly put them in your program. No need to be aware of or bother with BLOAD's, HIMEM:, etc.

Best of all, with just this starter package, you'll have the routines to put High Resolution **graphics** and **sound** in your programs immediately! Also included is a versatile **"Print Using"** module and powerful **search** and **sort** routines (for single dimension arrays: Search: 1000 elements in 1 second. Sort: 1000 elements in 90 seconds).

Additional library disks are already available. Some of the routines included in this package (plus others not listed) are:

SWAP	Swaps two string or numeric values.
TEXT OUTPUT	Prints with no "word break" on screen.
STRING INPUT	Input any string, regardless of commas, etc.
ERR	Stack fix for Applesoft ONERR handling.
GOTO	Allows computed GOTO's. Example: GOTO 5*X.
GOSUB	Allows computed GOSUBs.
MOVE	Block memory move routine.
RESET HANDLER	Treats RESET with ONERR; or will RUN or reboot disk.
HIRES ASCII	Character set for mixing text with Hires graphics.
TURTLE GRAPHICS	Versatile Hires graphics routines for easy drawing of Hires figures.

£39.95 + VAT

SHARP
First, and foremost

LEICESTER
computer centre limited

apple computer
Sales and Service

LEICESTER
computer centre limited
67 Regent Road, Leicester LE1 6YF. Tel: 0533 556268

Fareham

Dear Sir,

Special Interest Group

I have a special interest in Amateur Radio. My Call Sign is G3YMS.
I work SSB on HF and FM on VHF.
I would like to work RTTY on HF/VHF utilising the Apple as the terminal.
If there are other members with similar interests would they care to contact:-

John H Ison G3YMS
[illegible]

Yours truly

John

(Ed. Ah, G3YMS, if only you had purchased all the back numbers of Hard Core you would know that hams have been in BASUG from the start. See Derek Turner's article for the latest news of the airwaves).

Ealing

Dear Sirs,

The letter of Mr Ted Lepley in the last copy of 'Hard Core' causes me to write to you.
Three years ago I saw in one American electronics magazine a reference to the same kind of occurence and I could not really believe that this could happen here. Thanks mainly to 12 years in the Amateur radio fraternity I learned something about electronics and the following could be of help to your readers:
The Apple and the ITT microcomputer power supplies are very similar in design and both have hidden inside a small fuse. It is just a matter of opening the box and you'll see it (Now you know why 'they' do not like you to open it). I'm not saying that things are always so easy, but normally if your power went 'dead' without emitting some 'noise' it is 80 percent probable that the fuse has just blown (the Apple power supply is a switched mode type and the frequency tends to come to the audible range if you overload it). If that is the case replace it with one of the same rating.
I really do not understand why people are not told about that. It is very sad, to say the least. Anyway, next time you come to a dead Apple or ITT do please check the fuse first. You could be paying #170.00 for a 20p fuse.

Yours sincerely

C.P. Santos

EXPANDED HIRES PICTURES

By Richard Teed

This program should be used if you want a large screen dump or your printer does not have graphics.

As most printers do not print upwards of 192 columns the picture will have to be split at line 130.

The format of a hires screen is 8 lots of 1024 by 8 lots of 128 by 3 lots of 40: thus to see this try the following program:

```
100 FOR N=0 TO 80 STEP 40
110 FOR N1=N TO N+7*128
120 FOR N2=N1 to N1+7*1024
130 POKE N2,255
140 NEXT N2,N1,N
```

If your printer is unable to do upwards of 192 columns then line 130 of the program must be changed. If you want to do 3 drops of 64 columns then remove the loop at line 130 and do 3 runs with N1 set to:

```
8192+80+(7*128)+(7*1024) run 1
8192+40+(7*128)+(7*1024) run 2
8192+00+(7*128)+(7*1024) run 3
```

To do 2 drops - 1 of 128 columns and another of 64, do run 3 above for the final 64 columns and a run with U set to 8192+40+(7*128)+(7*1024)

Line 90 can probably also be removed. You will need the PR#1 but the CHR$(15) sets my printer to small characters and the rest of the line is to set the size of the line feed.

```
]LIST
10  DATA 0,0,0,173,1,64,45,0,64,1
    41,2,64,96
11  REM


12  REM  BEGIN  LDA $4001
13  REM         AND $4000
14  REM         STA $4002
            DO A BIT TEST. 4002
     IS GREATER THAN ZERO IF BIT
     IS ON
15  REM         RTS
           BACK TO BASIC
16  REM  TERM   END
17  REM
18  FOR  N = 16384  TO  16384 +
    12
19      READ  A: POKE  N,A

20  NEXT  N

21  PRINT  "INPUT YOUR MESSAGE "
    ;: LET  A$
30  INPUT "INPUT THE SIZE OF THE
    LINE FEED (4 IS AS PER THE G
    RAPHICS SCREEN, 7 IS FOR COM
    P-LETE CHARACTERS ";LF
40  PRINT  "TYPE 1 FOR INVERSE P
    RINTING ";: GET  AN$: PRINT
     AN$: POKE  - 16304,0: POKE
     - 16297,0
90  PR# 1: PRINT : POKE  1657,12
    8: PRINT  CHR$ (15): PRINT
     CHR$ (27)  +  CHR$ (65)  +
     CHR$ (LF)
100 A = 8192 + 80 + (7 * 128) + (
    7 * 1024):L = A:U = 8192 + 4
    0 + (7 * 128) + (7 * 1024)
110 B = A:C = A:D = 1024:E = 128:
    BIT = 1:HC = 0:MS = 1
120  FOR  N = 1  TO  40
121      FOR  BT = 1  TO  7
130          FOR  N1 = L  TO  U
    STEP  - 40
140              FOR  N2 = N1  TO
    N1  -  (7 * 128)  STEP  -
    128
150                  FOR  N3 = N
    2  TO  N2  -  (7 * 1024)  STEP
     - 1024
160                      P =  PEEK
    (N3): POKE  16384,P: POKE  1
    6385,BIT: CALL  16387:R =  PEEK
    (16386)
161                      IF  R =
    0  AND  AN$ < > "1"  THEN
     PRINT  " ";: GOTO  180
162                      IF  R >
    0  AND  AN$ = "1"  THEN  PRINT
     " ";: GOTO  180
170                      IF  R =
    0  AND  AN$ = "1"  THEN  PRINT
     MID$ (A$,MS,1);: GOTO  180
172                      PRINT
     MID$ (A$,MS,1);
180                      MS = MS
    + 1: IF  MS  >  LEN (A $)
    THEN  MS = 1
190                  NEXT  N3
191              NEXT  N2
192          NEXT  N1
200          PRINT :BIT = BIT  *
    2:MS = 1
201      NEXT  BT
210      L = L + 1 : U = U + 1 :
    BIT = 1
220  NEXT  N
230  PRINT  CHR$ (27) +  CHR$ (
    50)
240  PR# 0
```

C/WP's GREAT SUMMER PRINT-OUT

C/WP not only sell APPLE computers at prices so low they turn others green. We sell printers at ridiculous prices, too.

If your printer is off-colour buy a new one from C/WP, the micro-computer experts. Whatever make your computer you can benefit from our summer madness.

Or why not a little madness of your own and treat yourself to a brand new APPLE computer to run your printer. APPLE prices have never been lower. And ours are the lowest anywhere.

Whether you want a printer, an APPLE, or any of the hundreds of add-ons and peripherals we stock, you'll find it hard to find a keener price or more knowledgeable support.

Write or phone or just pop in for our full hardware and software list.

If you're hungry for a printer at these incredible prices, contact C/WP Computers on 01-630 7444.

C/WP PRINTER OFFER

	EX-VAT PRICES	
MATRIX PRINTERS	TYPICAL PRICE £	**C/WP PRICE £**
Seikosha GP100A 30cps, tractor	215	**149**
Microline 80, 80cps friction and tractor	235	**199**
Microline 82A, 120cps friction and tractor	350	**299**
Microline 83A, 120cps friction and tractor, 15.5 inch paper	575	**479**
Centronics 739, 100cps friction and tractor	529	**349**
Epson Type 3 (80 and 100)		PLEASE CALL US
DAISY WHEELS		
Olivetti Praxis, 11cps, KSR	480	**399**
TEC 40, 40cps	1285	**995**
TEC 55, 55cps	1675	**1290**
Qume 5/45, 45cps	1720	**1449**
Qume 9/35, 35cps, KSR	1483	**1383**
Diablo 630, 40cps plastic/metal wheel	1680	**1550**
Fujitsu SP830, 80cps plastic/metal wheel	1992	**1775**

Prices do not include cables or interfaces to computers. Apple computers parallel graphics or serial £70. Osborne £30. Limited offer to 30 September.

C/WP

C/WP Computers
108 Rochester Row, London SW1P 1JP
Telephone: 01-630 7444

HOW TO CALCULATE SAMPLE VARIANCE AND STANDARD DEVIATION ACCURATELY

By R.D. Purves

Sample variance and standard deviation are statistics which measure the extent to which individual observations are "scattered" about the arithmetic mean value. People wishing to calculate these elementary statistics often reach for an old-fashioned text-book in which they find instructions for the "desk calculator" method. In Applesoft, one might write:

```
100 SIGMX = 0.0:SSQ = 0.0:N = 0
110 PRINT "SAMPLE #" N + 1;
120 INPUT " <-9999 TO END> ? "; X
130 IF X = -9999 GOTO 160
140 SIGMX = SIGMX + X:SSQ = SSQ + X ^ 2
150 N = N + 1: GOTO 110
160 VAR = (SSQ - SIGMX ^ 2 / N)/(N - 1)
170 PRINT "MEAN = "SIGMX / N
180 PRINT "SAMPLE VARIANCE = "VAR
190 PRINT "STANDARD DEV. = " SQR (VAR)
```

This innocent looking program conceals some numerical horrors. It is algebraically correct (sample values 1.0, 2.0 & 3.0 give variance = 1.0), and does not make the mistake of forcing the user to count the number of data items to be entered (computers count better than people do). But it gives poor accuracy for tightly bunched data. Sample values 2000.0, 2000.1, 2000.2 give a variance of 7.8E-3, whereas the exact true value is 0.01. Where does this inaccuracy arise?

In line 160 SIGMX ^ 2 / N is subtracted from SSQ. If you print out these values you will see that for closely spaced data they are very nearly equal. Most of the leading digits are the same, and only a few trailing digits are available to convey numerical significance to the result of the subtraction. This "subtractive cancellation" is one of the main problems in numerical computation. In general there are three (not mutually exclusive) ways to deal with the difficulty.

(1) Use more accurate floating point arithmetic. To simplify a little, the accuracy of floating point operations is governed by the number of bytes of memory allotted to each variable. In IBM machines, single precision variables have 4 bytes allotted, the same as in Apple Pascal; numbers can be represented with a precision slightly better than 1 part in a million. Applesoft serves us better! 5 bytes are allotted, giving a 256-fold improvement in precision (this is an excellent reason for preferring Applesoft to Pascal). Hayden Book Co. sells a double precision software package for the Apple, giving a fabulous 21-digit precision, but this shot-gun approach to numerical accuracy is usually not needed, as we shall see.

(2) *Compute with* special care the two quantities to be subtracted. In the program example, squared quantities have been calculated by the power function X^2, which is neither accurate nor fast. Applesoft interprets X^2 as EXP (2 * LOG (X)). The relatively small error in the LOG routine is magnified by the subsequent exponentiation. If we replace the power functions by direct multiplication (X * X), the program performs a little better. However, an algorithm that is upset by errors in the 8th and 9th digits cannot be called a good one. This leads us to:

(3) Use an algorithm that avoids subtraction of nearly equal quantities. Two options are available for our variance calculation. For the first, suppose the N data items to have been gathered and stored in an array X().

```
160 SIGMX = 0.0:SSQ = 0.0
170 REM COMPUTE MEAN
180 FOR I = 1 TO N
190 SIGMX = SIGMX + X(I)
200 NEXT I
210 MEAN = SIGMX / N
220 REM COMPUTE VARIANCE
230 FOR I = 1 TO N
240 TEMP = MEAN - X(I)
250 SSQ = SSQ + TEMP * TEMP
260 NEXT I
270 VAR = SSQ / (N - 1)
```

Note that the vital subtraction (now in line 240) is of values much less nearly equal than in line 160 of the first listing. In consequence this version is hardly affected by cancellation, even for data which completely flummoxes the first program (e.g. 10000, 10000.1, 10000.2; variance = 0.01 exactly). If you cannot or do not want to store the data in an array, a cumulative processing method is:

```
100 SSQ = 0.0:MEAN = 0.0:N = 0
110 PRINT "SAMPLE #" N + 1;
120 INPUT " <-9999 TO END> ? ";X
130 IF X = -9999 GOTO 180
140 N = N + 1:DX = (X - MEAN) / N
150 REM UPDATE SUM-OF-SQUARES AND MEAN
160 SSQ = SSQ + DX * DX * N * (N - 1)
170 MEAN = MEAN + DX: GOTO 110
180 REM CALCULATE VARIANCE
190 VAR = SSQ / (N - 1)
200 PRINT "MEAN = "MEAN
210 PRINT "SAMPLE VARIANCE = "VAR
220 PRINT "STANDARD DEV. = " SQR (VAR)
```

This is still not a full working routine, as we have not provided any means for data validation and correction. As the textbooks put it, "that is left as an exercise for the reader".

NOW YOU SEE IT...

by Neil Lomas

Dr. G. A. Manson, in a short article in the June issue of Personal Computer World, describes how the Apple sometimes produces half-intensity dots on the high-res screen. According to Dr. Manson, the point x,y will be affected if

i) HCOLOR = 5, x is odd, and x+1 is divisible by 7

or ii) HCOLOR = 6, x is even, and x+1 is divisible by 7

or iii) HCOLOR = 7 and x+1 is divisible by 7.

The remedy, he suggests, is to set HCOLOR = 4 and plot the point x+1,y in black (or, more precisely, black2). Dr. Manson concludes that there is a fault in the Apple's plotting routines.

I had encountered a similar problem to this in a simple program I wrote a couple of months ago to produce pie-charts, graphs, and histograms from tables of values. This program mixed Applesoft and machine code and produced an effect which I could not logically explain in terms of hardware or software bugs. The article roused me into taking a fresh look at the problem, and I soon realised that there was something I had overlooked which produced a rational explanation.

The problem is at the same time simpler and more extensive than Dr. Manson describes. The eight bits in each byte of the screen memory map comprise 7 data bits (one bit corresponding to a screen dot) and a colour control bit. The setting of the colour control bit controls the colour set of the 7 dots. All this is described on page 19 of the Apple Reference Manual - if you have an ITT2020, this does not apply (but you won't suffer from the bug anyway!)

The problem occurs if the right-hand dot of a group of 7 is lit (bit 6 of the byte is set) and the colour control bit is different from that of the next byte. If the colour control bit of the lower (left-hand) byte is on and that of the next (right-hand) byte is off, the dot in question will be only half-lit (the problem described by Dr. Manson). If the settings of the colour control bits are reversed, however, not only will the dot in question be fully lit, but so will the dot immediately to the right of it, whether or not it is supposed to be. On some colour systems, a half-intensity dot will be a different colour and may not be visible from more than a couple of feet, while I have seen a monochrome system where the half-intensity dot was invisible from any distance.

If it is important that your displays do not exhibit either of these faults, there are three options available. The first option is always to use the "lower" (0-3) colour set, the second to use only the "upper" (4-7) set and to follow HGR or HGR2 with HCOLOR = 4 : HPLOT 0,0 : CALL 62454 (which initialises all the colour control bits). If you really must use both colour sets in the same display, the only way to avoid the problem is to ensure that the right-hand dot of a group of 7 is never lit if it is on a colour change boundary.

Dr. Manson's conclusion that the fault lies in the Apple's plotting routines is wrong, as can easily be shown by poking values directly into the memory. My own guess is that it is a timing problem in the conversion of memory contents to video signals, but as my knowledge of how this is done is nil, I may be equally wrong. Perhaps the people who freed ITT owners from behind bars can perform a similar service for the Apple users.

APPLE BOOKS

VISICALC USERS - you will know that we have already been advertising THE POWER OF VISICALC at £8.50

Now we can also offer:-

VISICALC HOME AND OFFICE COMPANION by Castlewitz, chisausky and Kronberg which has similar models for use in Investment, General Business, Inventory Control, Personal Finance, Household Aids etc. over 20 models all together.

Price including P&P ...£12.20

FOR THE 6502 machine code programmer the master of machine code programming books Lance Levental has teamed up with Winthrop Saville to produce a thick book of
6502 ASSEMBLY LANGUAGE SUBROUTINES

over 500 pages of all aspects of 6502 programming, from general methods to using arrays interrupts and I/O via arithmetic and string handling.

Price including P&P ...£12.50

ALL PRICES INCLUDE VAT UNLESS SPECIFIC MENTION IS MADE. ALL PRICES INCLUDE P&P

MICROSOURCE 1 Branch Road, Park Street St. Albans.
Tel: Park Street (0727) 72917

REVIEW OF TYPE-RIGHT LOWER CASE ADAPTOR

By Martin Rogers.

Being the owner of a pre-revision 7 Apple I was interested in the Microsource lower case conversion kit as so many of the lower case conversions available are only suitable for Revision 7 or later Apples. The Type-Right was designed by David Bolton, with software by Ian Trackman.
The conversion kit comprises a replacement Character Generator Chip, an assortment of adaptor boards, connectors and an isolation switch. Also included are a set of instructions for doing the actual conversion, a handbook explaining the use of Type-Right with Apple Writer and incorporating lower case letters in your Basic programs. Finally there is a diskette of software containing:-

- A conversion program that will apply suitable patches to the Apple Writer programs, to allow the use of English, French and German upper and lower case character sets.

- The source listings of the program patches, to allow the ambitious programmer to include his own enhancements.

- Machine Code subroutines which allow you to use Type-Right from BASIC programs with software controlled shift lock.

- The source listings of these subroutines.

- A demonstration program written in Applesoft BASIC demonstrating the techniques of using Type-Right.

It is most important that you get the correct conversion kit for your particular Apple II/ITT 2020. Fitting the wrong one could permanently damage both your computer and Type-Right. Microsource were very helpful on advising me of the correct version for my particular Apple.

Fitting might appear to be a daunting task to one who has no experience with electronics, and who has not had the top off his Apple since day one. However, following the fitting instructions which were designed to lead you through each step of the conversion proved to be quite straightforward, the whole job taking only half an hour from start to finish.

The first task involves the removal of all the peripheral cards and connectors. The importance of earthing any static electricity in your body is stressed, to avoid any damage to the components.

The case is then removed from the base of the computer by the removal of ten screws from the base. This was the only case of ambiguity in the instructions. The text mentioned screws labeled A,B and C on the diagram, but the diagram showed screws labeled 1 and 2. However it was perfectly obvious when you looked at the base of the Apple.

The keyboard connector is then unplugged to allow the the case to be put to one side.

The standard Character Generator chip is then removed. Luckily I had the I.C. puller from the DOS 3.3 conversion kit which removed the chip with ease. Failing this, the chip can be removed by gently prising each end with a small screwdriver. Another chip is then removed from the socket in front of the keyboard connector.

This chip is then inserted into an adaptor socket supplied with the kit, and then both are re-inserted into the original socket. A large adaptor board bearing the new Type-Right character generator and another chip is then inserted into the socket which formerly held the standard Apple character generator chip.

Another board containing the Shift Key adaptor is then inserted into the keyboard connector socket.

For Apples younger than mine, a probe is then attached to one of the wires going to the Encoder card on the bottom of the keyboard. For my ancient friend, a wire had to be soldered between two contacts on the keyboard printed circuit. This was the hardest part of the whole operation, taking great care not to apply too much heat or bridge any of the circuits, but was sucessfully accomplished.

The case is then lowered onto the base and the Keyboard connector is plugged into the new adaptor socket. The screws are then replaced to hold the base onto the case.

The Isolator switch must now be fitted so that it protrudes through one of the ventilation slots on the side of the case. As no bracket was supplied, I had to make one up out of a thin piece of aluminium which I attached to the case with sticky pads.

The only remaining job was to replace all the cards and connectors and to boot up the system after a last check that all was well.

The Demo program supplied on the diskette proved that all was well and that I did indeed

have very elegant lower case characters on the screen.

The next task was to apply the patches to Apple Writer so that lower case facilities could be enjoyed. After making a backup copy, the Apple Writer Patcher program was run. The program prompted for the Slot and Drive containing the Apple Writer disk and then gave the choice of the

English, French or German character sets. Having applied the patches to both the Editor and Printer programs the program terminates.

The revised Apple Writer program has the following differences.

- The Menu has L/C in the heading to remind you that you have the Type-Right version running.

- Normal key depressions now appear on the screen in true lower case.

- Upper case letters are now typed by pressing either of the two shift keys with the desired key, the letter appearing on the screen in upper case, rather than Inverse upper case.

- The shift key works normally on the shifted characters of all keys except for N and P. Formally these keys when shifted produced the characters @ and ^. To produce these characters now requires the depression of CTRL SHIFT P and N respectively.

- The ESC key now only requires one depression to enter edit mode as it is not used to select upper case.

- The Isolator switch can be used to disable Type-Right, and if you have a block of text to insert that is all in upper case, this is a handy way of doing it rather than holding down the shift key. Note that this does not re-establish the old Apple Writer ESC key functions.

- The missing ASCII characters [\ _ can now be typed from the keyboard using CTRL J G and O respectively. Four additional characters are now available. CTRL A L N Z give { | ~ } respectively.

- Whereas the characters mentioned above appear correctly on the screen, the installation does not guarantee that your printer will print them. The Centronics 737 and 739 printers will do so, but the use of Go-Between is recommended for proportional spacing.

Type-Right also caters for the inclusion of its facilities in BASIC programs. The Master diskette contains a selection of subroutines which can be incorporated in a BASIC program.

You have the facility to to incorporate either the English, French or German character sets.

The machine code subroutines are installed between DOS and the DOS buffers or in the case of the smaller English set, can alternatively be loaded at $300.

When typing in BASIC program statements you must of course use the normal upper case letters, but text in Input and Print statements, strings, data statements and Rems can be in lower case.

To turn on and off within the program is done by the use of three soft switches. Control S, the "temporary" command, turns on lower case, this mode being cancelled when Control S, X or Return is entered. Control W, the "permanent" command, remains in effect until Control A is entered. Alternatively, a series of POKE statements can be used to achieve the same results.

For experienced assembly language programmers, the source listings are available to enable you to modify the standard routines and assign any of the special characters on the chip to the keys of your choice.

The additional ASCII characters are again available, as with Apple Writer.

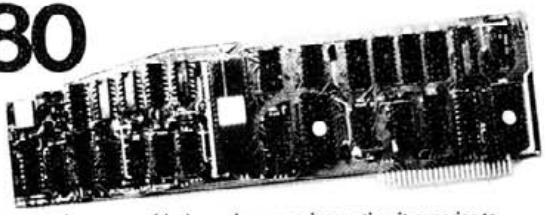
The keyboard editing facilities are different if Type-Right is enabled. (I find it easier to flip the Isolator switch!). The cursor control keys, Control A,B,C,D,I,J,K and M must now be shifted to operate.

There is a slight problem with the use of the PR# and IN# BASIC commands. Before they can be used, the Type-Right routines must be disconnected. This is achieved by incorporating the BASIC Unhooker and Loader routines in your program.

In conclusion, I have found the Type-Right kit to be a well thought out and professionally produced enhancement to the Apple II. It is easy to fit, following the comprehensive instructions. The Apple standard word processor, Apple Writer, is much easier to operate, particularly by non-Apple people, being much more like a standard typewriter in operation and appearance. For the BASIC programmer the package enables him to also make his application programs "user friendly". A worthwhile enhancement to the Apple!!

VISION-80

80 column card for Apple II Computer
R.R.P. **£195** + VAT
Dealer enquiries invited



The Vision-80 has achieved world renowned success with the major magazines rating it superior to all its competitors. "CALL-A.P.P.L.E." May, 1981 scored the card higher than any of the other cards reviewed and described the 9 x 11 dot character set with 3 dot descenders as superb. In the May '82 edition of "BYTE", the senior editor described the Vision-80 as "a very refined 80-column board", and was most impressed with the many features. "SOFT TALK" magazine has simply rated it as the best!

The Vision-80 fully supports the Applesoft commands HOME, GR, HGR, HGR2, TEXT, TAB, VTAB, HTAB, INVERSE, NORMAL, and the text window is fully supported according to Apple's conventions. No other 80-column card can do this. No mechanical switches are needed to change from 40-column to 80-column, or to low or high resolution graphics, giving 5 independent screens. The keyboard is transformed into a proper typewriter keyboard, providing upper and lower case entry, and allowing the shift keys to function correctly with a visual shift lock indicator.

In Pascal and CP/M, HIGHLIGHT, LOWLIGHT, FLUSH, function KEYPRESS, and the type ahead buffer are all supported. The Vision-80 works with all languages available for Apple II and such software products as ZARDAX, EPSILON (Sandy's W.P.) and WORDSTAR (CP/M) word processors. In addition, a unique and powerful communications facility is present in hardware on the board. Files may be automatically uploaded or downloaded from disc to a distant computer, or to the printer. Also it allows the computer and disc to be controlled remotely from another computer or terminal.

VISION-80 VC This is a utility that works with all versions of VISICALC 3.3 to provide 80 columns with the VISION-80. It allows entry of Upper/Lower case, inverse and normal, and has an extremely clean display (unlike the competition!).

(NOTE: you need no switch plates, inverse chips or keyboard enhancers.)
#35 + VAT)

ROSETTA DOS/PASCAL utility. This superb Australian program is written in PASCAL and allows movement of any sort of DOS or PASCAL programs in any direction. A library unit called OUASIDOS is included for the user to link to other PASCAL programs so PASCAL can read and write to DOS discs.
#35 + VAT

COMING SOON Memory expansion for the VISION-80 VC.

Modification for APPLEWRITER II to work with VISION-80.

PYNWON COMPUTER SERVICES

17 Watermill Lane Edmonton London N18 1SU
01-884 0879

& Pam Computers Pete & Pam Computers Pete & Pam Compu

VISICALC 80 COLUMN DISPLAY on an APPLE II !

Yes, now its possible for all VISICALC users to obtain Screen Display in 80 COLUMNS, and to have additional memory available for VISICALC applications !!

Install one Saturn 128K Board and get 145K for Visicale

Install one Saturn 32K Board and get 49K for Visicale

Install one Saturn 128K Board plus one 32K Board = 177K for Visicale

VC EXPAND 80 TO GIVE ADDITIONAL MEMORY AND 80 COLUMN DISPLAY – £69.00

VC EXPAND WITHOUT 80 COLUMN DISPLAY – £55.00

VIDEX VISICALC 80 COLUMN SOFTWARE (No Memory Expansion) – £29.95

SATURN 128K BOARD – £359 SATURN 32K BOARD – £149

VIDEX VIDEO TERM – £195 VIDEX 40/80 Col Software Controlled Switch – £19.95

NEW PACKAGING From VISICORP

* VISICALC	£159.00
* VISIFILE	£159.00
* VISITERM	£79.00
* DESKTOP/PLAN	£159.00
* VISIDEX	£159.00
* VISIPLOT	£125.00
* VISITREND/PLOT	£180.00
* VISISCHEDULE	£195.00
* VISISCHEDULE III	£195.00
* DESKTOP/PLAN III	£195.00

NEW **VISICALC BUSINESS FORECASTING MODEL**

An easy to use VisiCalc application package for financial analysis and forecasting, containing 7 ready to use worksheets

£75.00

TRIPLE YOUR DISK ACCESS SPEED

No hardware modification required.

FastDOS

Fast Disk operating system for APPLE][computers.

Completely compatible with DOS disks.

Loads and saves standard DOS files.

Completely compatible with all DOS/APPLESOFT programs that access DOS through standard hooks, including FID and MUFFIN.

Executes all standard DOS commands.

Comparative timings:	DOS	FDOS
Bloading integer basic	13 sec	3 sec
Cataloging a 12 file disk	2 sec	1 sec
Saving a 10 sector program	6 sec	2 sec
Saving a 100 sector program	34 sec	7 sec
Loading a 100 sector program	24 sec	7 sec

Requires 48K

Recommended Retail Price £19.95

UTILITIES

BACK IT UP BIT COPIER	£39.95
SUPER DISK COPY III	£17.95
DISK RECOVERY	£17.95
DISK ORGANISER II	£17.95
MULTI DISK CATALOG III	£15.95
DOS PLUS	£15.95
QUICKLOADER	£15.95
APPLESOFT PROGRAM OPTIMISER	£13.95
MACRO SCREEN EDITOR	£29.95
MON & DISK	£21.95
ACE (Applesoft Command Editor)	£21.95
LIST MASTER	£22.95
DAKIN 5 PROGRAMMING AIDS 3.3	£49.95
HIGHER TEXT II	£21.95
HIGHER GRAPHICS II	£18.95
HIGHER FONTS I	£8.95
PROGRAM LINE EDITOR	£21.95
MICROSOFT A.L.D.S.	£79.00
MICROSOFT TASC - The APPLE Compiler	£109.00
MICROSOFT APPLE Z80 - Basic Compiler	£209.95

NEW **BAG OF TRICKS**

From the authors of Beneath Apple DOS

Includes many "hand holding" tutorials that assist you in repairing damaged diskettes and allow you to change sector ordering, reconstruct blown catalogs, etc.

£21.95

SYNERGIZER with FREE SUPERCALC contains

Z-Card Z-80 Processor Card

C/PM OP System and Licence

Rodney Zaks C/PM manual from Sybex

16K ADD RAM Ramcard

Smarterm 80 col card with enhanced CHR set and integral soft switch together with free Supercalc

SUMMER SPECIAL ! £399.00

* *Without 16K CARD* £335.00

Z-CARD WITH SUPERCALC and CP/M

SPECIAL PRICE £199.00

Pete & Pam Computers

Mail Order & Distribution:
Waingate Lodge, Waingate Close,
Rossendale, BB4 7SQ
Phone: (0706) 227011
Telex: 635740 Petpam G

London Retail:
103-5 Blegborough Road,
London, SW16 6DL
Phones: 01-769 1022/3/4

Norwegian Agent:
The Norwegian Software House
Address Okernveien 145
Oslo 5
Telephone (02) 22 89 78

BOOKS (No VAT)

VISICALC Home & Office Companion	£11.50
APPLE BASIC : Data File Programming	£8.95
What's Where in the APPLE?	£8.95
Science & Engineering Programs APPLE II	£11.50
A Guide to Programming in APPLE SOFT	£11.00
APPLE Pascal Games	£11.45
PASCAL PROGRAMMING for APPLE	£10.45
APPLE Pascal - a hands-on approach	£10.50
Osborne CP/M User Guide	£9.95

MACHINE COVERS - only the best material used

Apple only	£5.95
Single Disk	£2.95
2 Stacked Disks	£4.45
Apple, 2 Disks + 9" Monitor or Apple + 12" Monitor	£8.95
Apple + 2 Disks	£7.95
Epson MX 70/80	£5.45
Paper Tiger 445 - 460	£5.45
9" Monitor	£4.95
Apple, 2 Disks + 12" Monitor	£9.50
Hitachi 12" Cover	£7.50
Qume Sprint 5 Cover	£10.95
Apple III Cover inc. Monitor III	£12.95
Sirius Machine and Monitor Cover	£12.95
Sirius Keyboard Cover	£4.45
Epson MX 100	£7.45
NEC 12" Monitor	£7.50

THIS MONTH'S SPECIALS

CALCSTAR For APPLE	
WORDSTAR SYSTEMS	£59.00
OSBORNE C/PM USER GUIDE (Book – No VAT)	£9.95
BATTLE OF SHILOH	£24.95
MATHEMAGIC	£49.95
EXPEDITER II	£39.95
UZ80 Processor Card	£69.00
DIGITEK Colour Card	£89.00
16K RAM CARD	£65.00

A BIG HIT AT APPLE '82

timekit

The amazingly compact MICROWATCH real time clockcard and ELECTRONIC DIARY software for your APPLE II computer

£59.00

Prices do not include VAT please add 15% to your remittance Postage and Packing FREE